CHAPTER 7

Application of Fuzzy Rule Base Design Method

Peter Grabusts

In many classification tasks the final goal is usually to determine classes of objects. The final goal of fuzzy clustering is also the distribution of elements with highest membership functions into classes. The key issue is the possibility of extracting fuzzy rules that describe clustering results. The paper develops a method of fuzzy rule base designing for the numerical data, which enables extracting fuzzy rules in the form *IF-THEN*. To obtain the membership functions, the fuzzy c-means clustering algorithm is employed. The described methodology of fuzzy rule base designing allows one to classify the data. The practical part contains implementation examples.

Keywords - Fuzzy rule-based systems, clustering, system design

7.1 Introduction

In many situations it is possible to model the system behavior qualitatively using the expert's knowledge about the system based on the input and output data of

Peter Grabusts
Faculty of Engineering
Rezekne Academy of Technologies
Rezekne, Latvia
e-mail: peter@ru.lv

Editors: G.A. Papakostas, A.G. Hatzimichailidis, V.G. Kaburlasos, *Handbook of Fuzzy Sets Comparison - Theory, Algorithms and Applications*
DOI: 10.15579/gcsr.vol6.ch7, GCSR Vol. 6, pp. 149-168, 2016

the system. The data could be obtained using perfect mathematical modeling, from expert's knowledge or from experimental analysis of the system. As these data can also contain uncertainty, selecting a suitable approach is essential in order to include any information about the system. In such situations, the system identification can be done by using a fuzzy clustering technique, which involves grouping of data into fuzzy clusters of similar behavior, and translation of these clusters into *IF-THEN* fuzzy rules.

Clustering [5] is one of the most fundamental issues in pattern recognition. It plays a significant role in searching for structures in data. Given a finite set of data X, the problem of clustering in X is to find several cluster centres that can properly characterize relevant classes of X. In classic cluster analysis, these classes are required to form a partition of X with a strong degree of association for data within blocks of the partition and with a weak degree of association for data in different blocks. However, this requirement is too strong in many practical applications and it is thus desirable to replace it with a weaker requirement. When the requirement of a crisp partition of X is replaced with a weaker requirement of a fuzzy partition or a fuzzy pseudopartition on X, we refer to the emerging problem area as fuzzy clustering. Fuzzy pseudopartitions are often called fuzzy c-partitions, where c designates the number of fuzzy classes in the partition.

There are some basic methods of fuzzy clustering. One of them, based on fuzzy c-partitions, is called a *fuzzy c-means clustering* algorithm (FCM).

The main objective of the study was to extract fuzzy rules from the numerical data. The rules obtained constitute a specific knowledge base. Methods of fuzzy rule base design are widely used in different control processes. They, however, can be adapted to rule extraction from the numerical data. To accomplish this aim, input data have to be clustered. The paper employs a well-known FCM algorithm. As a result of clustering, data characterizing membership functions were derived.

The chapter describes the method employed to acquire rules in the *IF-THEN* in five stages. The stages are illustrated with examples that characterize the essence of the method. The experimental part contains an implementation example using the IRIS [4] data set. By employing the method of rule base design at different initially set values of membership function count, fuzzy rules were obtained through clustering. The performed experiments led to the conclusion that the acquired rules correctly describe the data and thus a rule base is generated. The extracted rules can help discover and then analyze the hidden knowledge in data sets.

7.2 Brief Review of Related Works

Cluster analysis is one of the basic techniques applied in the data analysis. The classical (hard) clustering methods limit the belonging of each point of the data set to exactly one cluster [5].

Fuzzy set theory proposed by [21] gave the idea of the uncertainty of belonging which was described by a membership function [11, 18, 20, 19]. The use of fuzzy sets provides imprecise class membership information. Application of fuzzy set theory in cluster analysis was early described in the work of Bellman, Kalaba and Zadeh [2] and

Ruspini [13].

According to the literature in the area of fuzzy clustering, the FCM clustering algorithms defined by Dunn [6] and generated by Bezdek [3] are the well-known and powerful methods applied in fuzzy cluster analysis. Nowadays classical fuzzy clustering algorithms have been widely studied. Even now, there are on-going studies on the application of the FCM algorithm for the needs of various sectors [1, 12].

Obtaining the rules as a result of fuzzy clustering is widely represented in bibliographic sources. One of the first who fully described it [10]: fuzzy clustering offers various possibilities for learning fuzzy *IF-THEN* rules from data for classification tasks as well as for function approximation problems like in fuzzy control.

The author under consideration has made a great contribution in obtaining the rules from fuzzy clustering [8]. The given book offers timely and important introduction to fuzzy cluster analysis, its methods and areas of use. The book systematically describes the various fuzzy clustering techniques so it is possible to choose the method that is the most appropriate for solving the problems. There is a good and very comprehensive review of the literature on the subject of research, pattern recognition, data analysis and rules output.

Analytical evaluation of the FCM and fuzzy system models is given in this work [15]. It can be concluded that studies on obtaining the rules with the help of FCM algorithm are being actively conducted, which has a significant impact on the emergence of new knowledge as a result of the use of the methodology.

7.3 System Identification Using Fuzzy Clustering

In general, the identification of the system involves structure identification and parameter identification. The structure identification consists of initial rule generation after elimination of insignificant variables in the form of *IF-THEN* rules and their fuzzy sets. Parameter identification includes consequent parameter identification based on certain objective criteria.

The model proposed by Takagi and Sugeno [14] is called TS fuzzy model, the consequent part is expressed as a linear combination of antecedents. In TS model the system with N rules and m antecedents can be expressed as:

$$R^1 : IF\, x_1\, is\, A_1^1\, and\, x_2\, is\, A_2^1\, and \ldots and\, x_m\, is\, A_m^1\, THEN\, y^1 = P_0^1 + P_1^1 x_1 + \ldots + P_m^1 x_m$$

$$\ldots$$

$$R^N : IF\, x_1\, is\, A_1^N\, and\, x_2\, is\, A_2^N\, and \ldots and\, x_m\, is\, A_m^N\, THEN\, y^N = P_0^N + P_1^N x_1 + \ldots + P_m^N x_m$$

where x_i is the i-th antecedent $(i = 0, 1, ..., m)$, R^j and y^j represent the j-th rule and its consequent $(j = 1, ..., N)$, respectively and P_i^j are the consequent parameters.

When input-output data are available *a priori*, fuzzy clustering is a technique that can be used for structure identification. Then, the consequent parameters can be optimized by the least square estimation (LSE) given by Takagi and Sugeno.

Figure 7.1: Example with two clusters. Cluster centres are marked with solid circles.

The identification of the system using fuzzy clustering involves formation of clusters in the data space and translation of these clusters into TSK rules so that the model obtained is close to the system being identified.

The FCM clustering algorithm, which has been widely studied and applied, needs *a priori* knowledge of the number of clusters. Whenever FCM requires a desired number of clusters and initial guess positions for each cluster center, the output rules strongly depend on the choice of initial values as the FCM algorithm iteratively forms a suitable cluster pattern in order to minimize an objective function dependent of cluster locations.

For example, we have the feature space with two clusters (see Fig.(7.1)):

The plot of the clusters in Fig.(7.1) suggests a relation between the variable x on the horizontal axis and y on the vertical axis. For example, the cluster in the upper right hand corner of the plot indicates, in very loose terms, that whenever x is "high", defined as near the right end of the horizontal axis, then y is also "high", defined as near the top end of the vertical axis. The relation can be described by the rule:

$$IF\, x\, is\, high\, THEN\, y\, is\, high$$

It seems possible to make some intuitive definitions of the two instances of the word "high" in the rule, based on the location of the cluster centre. The cluster in the lower left part of the Fig.(7.1) could be described as:

$$IF\, x\, is\, low\, THEN\, y\, is\, low$$

7.4 Fuzzy C-Means Clustering

The classical *c-means algorithm* [5] tries to locate clusters in the multi-dimensional feature space. The goal is to assign each point in the feature space to a particular cluster. The basic approach is as follows:

1. Manually seek the algorithm with c cluster centres, one for each cluster we are seeking. This requires prior information from the outside world about the number of different clusters into which the points are to be divided; thus the algorithm belongs to the class of supervised algorithms.

2. Each point is assigned to the closest cluster centre to it.

3. A new cluster centre is computed for each class by taking the mean values of the coordinates of the points assigned to it.

4. If not finished according to some stopping criterion, go to step 2.

Formally, the *c-means algorithm* finds a centre in each cluster, minimizing an objective function of a distance measure. The objective function depends on the distances between vectors u_k and cluster centres c_i, and when the Euclidean distance is chosen as a distance function, the expression for the objective function is:

$$J = \sum_{i=1}^{c} J_i = \sum_{i=1}^{c} \left(\sum_{k,u_k \in C_i} \|u_k - c_i\|^2 \right) \tag{7.1}$$

where J_i is the objective function within cluster i.

The partitioned clusters are typically defined by a $c \times K$ binary characteristic matrix M, called the membership matrix, where each element m_{ik} is 1 if the k-th data point u_k belongs to cluster i, and 0 otherwise. Since a data point can only belong to one cluster, the membership matrix M has these properties:

P1: the sum of each column is one.

P2: the sum of all elements is K.

If the cluster centres c_i are fixed, the m_{ik} that minimise J_i can be derived as

$$m_{ik} = \begin{cases} 1, & \text{if } \|u_k - c_i\|^2 \le \|u_k - c_j\|^2 \\ 0, & \text{otherwise} \end{cases}, \forall j \ne i. \tag{7.2}$$

That is, u_k belongs to cluster i if c_i is the closest centre among all centres. If, on the other hand, m_{ik} is fixed, then the optimal centre c_i that minimises Eq.(7.3) is the mean of all vectors in cluster i:

$$c_i = \frac{1}{|C_i|} \sum_{k,u_k \in C_i} u_k. \tag{7.3}$$

where $|C_i|$ is the number of objects in C_i, and the summation is an element-by-element summation of vectors.

The algorithm is iterative, and there is no guarantee that it will converge to an optimal solution. The performance depends on the initial positions of the cluster centres, and it is advisable to employ certain method to find good initial cluster centres. It is also possible to initialize a random membership matrix M first and then follow the iterative procedure.

It is reasonable to assume that points between the two cluster centres, have a gradual membership of both clusters. Naturally this is accommodated by fuzzifying the definitions of "low" and "high". The FCM algorithm allows each data point to belong to a cluster to a degree specified by a membership grade, and thus each point may belong to several clusters.

The FCM algorithm partitions a collection of K data points specified by m-dimensional vectors u_k $(k = 1, 2, ..., K)$ into c fuzzy clusters, and finds a cluster centre in each,

Algorithm 7.1 [9]

The hard *c-means* algorithm has five steps:

1. Initialize the cluster centres c_i $(i = 1, 2, ..., c)$. This is typically achieved by randomly selecting c points from the data points.
2. Determine the membership matrix M by Eq.(7.2).
3. Compute the objective function Eq.(7.1). Stop if either it is below a certain threshold value, or its improvement over the previous iteration is below a certain tolerance.
4. Update the cluster centres according to Eq.(7.3).
5. Go to step 2.

minimizing an objective function. FCM is different from hard *c-means*, mainly because it employs fuzzy partitioning, where a point can belong to several clusters with degrees of membership. To accommodate the fuzzy partitioning, the membership matrix M is allowed to have elements in the range $[0, 1]$. A point's total membership of all clusters, however, must always be equal to unity maintaining the above mentioned properties (P1, P2) of the M matrix. The objective function is a generalization of Eq.(7.1):

$$J(M, c_1, c_2, ...c_c) = \sum_{i=1}^{c} J_i = \sum_{i=1}^{c} \sum_{k=1}^{K} m_{ik}^q d_{ik}^2 \tag{7.4}$$

where m_{ik} is a membership between 0 and 1, c_i is the centre of fuzzy cluster i, $d_{ik} = ||u_k - c_i||$ is the Euclidean distance between the i-th cluster centre and k-th point, $q \in [1, \infty)$ is a weighting exponent.

There are two necessary conditions for J to reach a minimum:

$$c_i = \frac{\sum_{k=1}^{K} m_{ik}^q u_k}{\sum_{k=1}^{K} m_{ik}^q} \tag{7.5}$$

and

$$m_{ik} = \frac{1}{\sum_{j=1}^{c} \left(\frac{d_{ik}}{d_{jk}}\right)^{2/(q-1)}} \tag{7.6}$$

The algorithm is simply an iteration through the preceding two conditions.

Alternatively, the cluster centres can be initialized first, before carrying out the iterative procedure. The algorithm may not converge to an optimum solution and the performance depends on the initial cluster centres, just as in the case of the hard *c-means* algorithm.

To demonstrate the FCM algorithm facilities, the following data set has been selected (see Table 7.1).

Algorithm 7.2 [9]

In a batch mode operation, the FCM algorithm determines the cluster centres c_i and the membership matrix M using the following steps:

1. Initialize the membership matrix M with random values between 0 and 1 within the constraints of P1 and P2.
2. Calculate c cluster centres c_i $(i = 1, 2, ..., c)$using Eq.(7.5).
3. Compute the objective function according to Eq.(7.4). Stop if either it is below a certain threshold level or its improvement over the previous iteration is below a certain tolerance.
4. Compute a new M using Eq.(7.6).
5. Go to step 2.

Table 7.1: Training set.

$X1$	-1.31	-0.64	0.36	1.69	-0.98	-0.98	0.02	0.36	-0.31	1.02	1.02	-0.31	1.36	-1.31
$X2$	-0.63	-0.21	-1.47	0.63	-0.63	1.47	0.21	0.21	-0.63	0.63	-0.63	1.89	-1.47	0.63

By means of the FCM algorithm the following cluster centres have been derived (see Fig.(7.2)).

The initial cluster centres were generated arbitrarily, whereas the final ones were formed as a result of the FCM algorithm execution. In accordance with the algorithm, objective function values were computed by Eq.(7.4) and membership matrix M was calculated by Eq.(7.6). Membership function distribution for two clusters is shown in Fig.(7.3).

In further experiments an attempt was made to enlarge the number of clusters. The following objective function values were derived:

2-clusters: Objective function = 17.75 (2 iterations)

3-clusters: Objective function = 9.47 (3 iterations)

4-clusters: Objective function = 5.02 (5 iterations).

The results are shown in Fig.(7.4).

It can be concluded that the FCM algorithm can be successfully applied in fuzzy clustering and its use is one of the preconditions for rule base extraction.

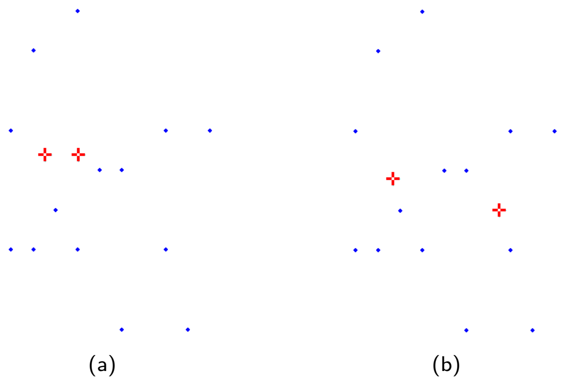


Figure 7.2: Initial (a) and final (b) cluster centres.

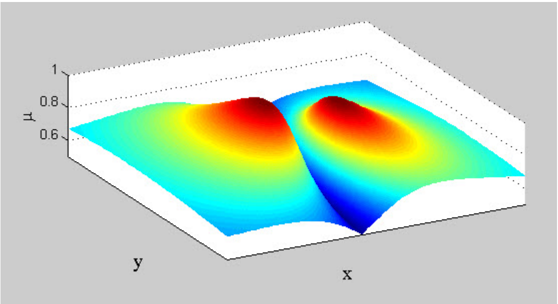
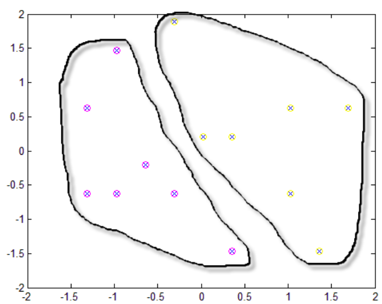
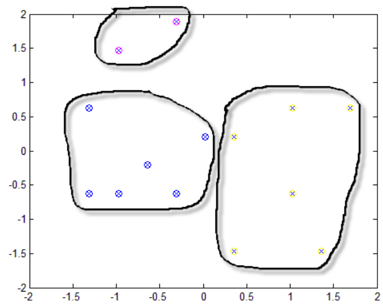


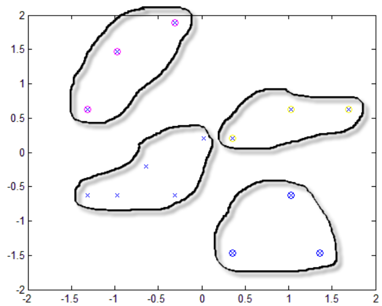
Figure 7.3: Membership functions for two clusters.



(a)



(b)



(c)

Figure 7.4: FCM algorithm results: (a) two, (b) three and (c) four clusters.

Table 7.2: Artificial data set.

$X1$	0.14	0.28	0.42	0.57	0.71	0.85
$X2$	0.85	0.42	0.71	0.28	0.57	0.14

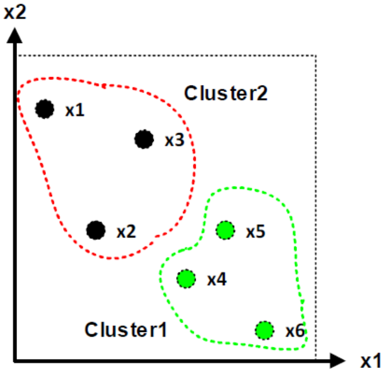


Figure 7.5: Data distribution in clusters.

7.5 Rules Acquisition with the Help of Fuzzy Clustering

7.5.1 Membership Matrix Transformation into Membership Functions

Further, an attempt was made to expand the application scope of FCM algorithm and obtain data characterizing the regularities in the form of the rule. All observations were made based on the data samples provided in Table 7.2.

After the cluster center initialization using FCM clustering algorithm ($c = 2$, the maximum number of iterations = 100), two clusters were found, calculated according to Eq.(7.5), membership matrix U and membership functions distribution were obtained using Eq.(7.6) (see. Fig.(7.5) and Fig.(7.6)).

The cluster centers $V = \{(0.6910; 0.2991), (0.2991; 0.6908)\}$ and membership ma-

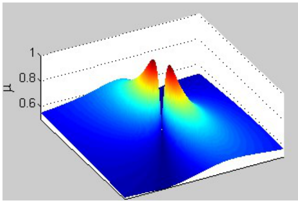


Figure 7.6: Distribution of membership functions in clusters.

Table 7.3: Distribution of membership functions into classes.

$X_1\,(MF_1)$	$X_1\,(MF_2)$	$X_2\,(MF_1)$	$X_2\,(MF_2)$	Clusters	
1.0000	0	0	1.0000	0	1
0.8028	0.1972	0.6056	0.3944	0	1
0.6056	0.3944	0.1972	0.8028	0	1
0.3944	0.6056	0.8028	0.1972	1	0
0.1972	0.8028	0.3944	0.6056	1	0
0	1.0000	1.0000	0	1	0

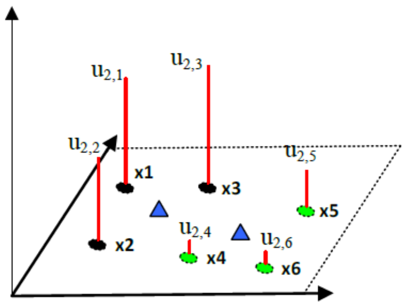


Figure 7.7: The membership function elements $u_{2,k}$and cluster centers.

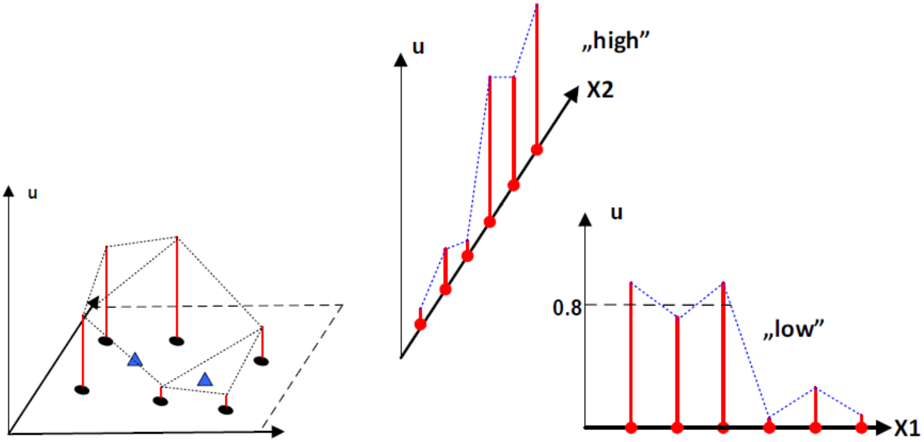
trix U (with two membership functions), as shown in Table 7.3, were obtained.

As can be seen from the table, Cluster 1 contains points x_4, x_5, x_6, whereas points x_1, x_2 and x_3 correspond to Cluster 2.

Figure 7.7 shows the cluster centers and the projection of the membership function elements on the u axis.

Furthermore, it was assumed that the membership functions $\mu_i(x_k) = u_{ik}$ for all $i = 1, 2$ and $k = 1, ..., 6$. There was a tendency to make the membership functions with the help of linguistic variables, which makes it easier to interpret the fuzzy systems. Terms such as "high", "low" or "medium" may well describe the one-dimensional position of the object. In case they are multi dimensional objects, it no longer is so easy. For example, how could the membership functions shown in Fig.(7.8) be described? Projection method is a common technique in fuzzy set theory [8]. The values of the elements in membership functions u_{ik} are projected to the coordinate axes, resulting in the profile shown in Fig.(7.8b).

Thus, the membership functions in the coordinate axis x_1 can be interpreted by the terms "low" or "x_1 is low". Accordingly, the functions in the coordinate axis x_2 can be interpreted as "high" or "x_2 is high". This method is called the cylindrical extension that specifies that the multi-dimensional data vector is used as the scalar argument of membership function in the projection space. The membership functions are extended with other dimensions as profile. In this way the original clusters with cylindrical extension profile conjunction - x_1 can be described as "low" and x_2 as "high". Projection method enables to approximate the fuzzy sets with convex membership functions, as



(a) (b)

Figure 7.8: Interpretation of (a) membership functions and (b) projection profile.

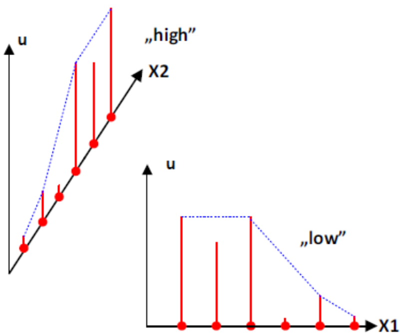


Figure 7.9: Membership function profiles after approximation.

it is shown in Fig.(7.9).

7.5.2 The Essence of Fuzzy Classifiers

In many classification tasks the goal is usually the definition of class objects. The goal of fuzzy clustering is also the distribution of elements with a higher membership functions into classes. Very important is the question of the possibility to get the fuzzy rules describing the clustering results. In two-dimensional case, each rule is associated with two rectangular intervals which characterize the *IF* conditional part function of the describing rule. Figure 7.10 illustrates the essence of the two-dimensional classification.

Classification problems in the simplest case (two-dimensional space) for two classes, separated with the help of a function, can be solved by using the fuzzy classification rules [20].

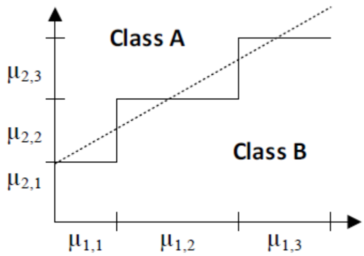


Figure 7.10: Essence of linearly separable classification.

To describe a fuzzy classification problem, the following assumptions are made. Assume, there exist p variables $x_1, x_2, ..., x_p$, which are defined in the interval $X_i = [a_i, b_i]$, $a_i < b_i$. The final class set C is given for which the following distribution is valid:

$$class: \; X_1 \times X_2 \times ...X_p \to C.$$

The objective is to find a classifier that could solve classification problem [8]. The fuzzy classifier is based on the set of final rules R for which the following holds:

$$R: \; \text{If } x_1 \text{ is } \mu_R^{(1)} \text{ and...and } x_p \text{ is } \mu_R^{(p)} \text{ Then } class \text{ is } C_R.$$

$C_R \in C$ is one of the classes. The $\mu_R^{(i)}$ are assumed to be fuzzy sets on X_i, i.e. $\mu_R^{(i)} : X_i \to [0, 1]$. Fuzzy sets $\mu_R^{(i)}$ are directly included in the rule. In real situations they can be replaced by the corresponding linguistic variables. Actually, input data vector is ascribed to class C if fuzzy rules determine vector's higher membership in class C. In [8] it is shown how 2 rules are obtained in the two-dimensional case:

$$\text{If } x \text{ is } \mu_1 \text{ and } y \text{ is } v_1 \text{ Then } class \text{ is } N$$
$$\text{If } x \text{ is } \mu_2 \text{ and } y \text{ is } v_2 \text{ Then } class \text{ is } P$$

7.5.3 Designing a Fuzzy Rule Base

In solving many practical applications, the information necessary for the development and implementation of a fuzzy system can be divided into two kinds: numerical (the result of measurements) and linguistic (obtained from experts). Most of fuzzy systems are implemented using the second kind of knowledge, which is mostly represented in the form of a fuzzy rule base. In cases when a fuzzy system with the numerical data has to be developed, certain important problems appear. A possible way to solve them is to use the neural-fuzzy systems, where neural networks are employed for rule base optimization. As a disadvantage of these systems, a long lasting iterative learning algorithm can be mentioned. In what follows, a method for knowledge base construction from the numerical data, proposed in [16, 17], is discussed.

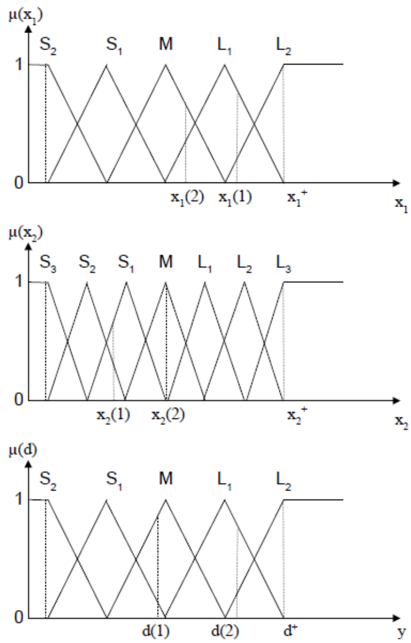


Figure 7.11: Partition of input and output data into intervals and corresponding membership functions.

For clarity purposes let us assume that a fuzzy system with two inputs (input signals) and one output is being constructed. Thus the following form of learning data is required:

$$(x_1(i), x_2(i), d(i)),\ \ i = 1, 2, ...$$

where $x_1(i)$ and $x_2(i)$ denote the incoming data but $d(i)$ is the expected output. The task of the system is to form fuzzy rules so that possibly the best result would be obtained on the output. The task stated can be accomplished in five stages.

Stage 1 - Separation of input and output data

Actually, the minimal and maximal values of the input data are known, so intervals are determined, in which the allowable values are located: . Each of the intervals is divided into $(2N + 1)$ parts. For particular interval parts linguistic variables can be set, for example, S_N(small N),...,S_1(small 1), M(middle), L_1(large 1),...,L_N(large N) and their membership functions can be determined. Figure 7.11 shows an example of similar distribution, where the domain of signal x_1 is divided into 5 subintervals $(N = 2)$, the domain of signal x_2 is divided into 7 subintervals $(N = 3)$ but the domain of the output signal is partitioned into 5 subintervals $(N = 2)$.

Stage 2 - Construction of fuzzy rules using learning set data

At this stage, membership degrees of learning data $(x_1(i), x_2(i))$ and $d(i)$ have to be determined for each of the selected domains. This is expressed using the values of membership functions. For example, in Fig.(7.11) the membership degree of $x_1(1)$ in domain L_1 is 0.8, in domain $L_2 - 0.2$; membership in other domains is 0. Similarly, the membership degree of $x_2(2)$ in domain M is 1, whereas its membership in other domains is 0. In the same way we will ascribe $x_1(i), x_2(i)$ and $d(i)$ to those domains where they have maximal membership degrees. Say, $x_1(1)$ has maximal membership degree in domain L_1, whereas $x_2(2)$ in domain M. Thus for each pair of learning data a single rule can be set, for example, in this way:

$$(x_1(1), x_2(1); d(1)) \to \{x_1(1)\,[max: 0.8 \text{ in domain } L_1]\,,$$
$$x_2(1)\,[max: 0.6 \text{ in domain } S_1]\,; d(1)\,[max: 0.9 \text{ in domain } M]\} \to$$
$$R^1: \text{If } (x_1 \text{ is } L_1 \text{ and } x_2 \text{ is } S_1) \text{ Then } y \text{ is } M.$$

Stage 3 - Determination of confidence degree for each rule

Taking into account that a lot of learning data pairs exist and for each of them a single rule can be generated, a possibility exists that the rules might be inconsistent. This relates to the rules having the same condition, but different conclusions. One of possible solutions to this problem might be assigning the confidence degree to each rule with a view to further choose the rule with the highest confidence degree. As a result, not only the problem of rule contradiction would be solved but also the total number of rules would decrease essentially.

For the rule of the form $R: \text{If } (x_1 \text{ is } A_1 \text{ and } x_2 \text{ is } A_2) \text{ Then } (y \text{ is } B)$ the confidence degree will be defined as follows:

$$SP\,(R) = \mu_{A_1}\,(x_1) \times \mu_{A_2}\,(x_2) \times \mu_B\,(y)\,.$$

Thus rule R^1 from the above example will have this confidence degree:

$$SP\left(R^1\right) = \mu_{L_1}\,(x_1) \times \mu_{M_1}\,(x_2) \times \mu_V\,(y) = 0.8 \times 0.6 \times 0.9 = 0.432.$$

Stage 4 - Formation of fuzzy rule base

The principle of fuzzy rule formation is shown in Fig.(7.12).

The rule base is set in the form of a table, which is completed with rules as follows. If a rule is given in the form $R^1: \text{If } (x_1 \text{ is } L_1 \text{ and } x \text{ is } S_1)$, $\text{Then } y \text{ is } M$, the value of a fuzzy set that is contained in the *Then* part of the rule, i.e. the value M in this example, is recorded in the point of intersection of column L_1 and row S_1. In case various rules with the same condition exist, a rule with the highest confidence degree is selected out of them.

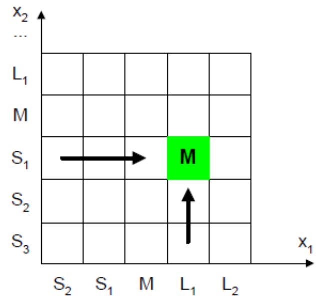


Figure 7.12: Formation of fuzzy rule base.

Stage 5 - Defuzzification

At this stage, mapping $f : (x_1, x_2) \to \bar{y}$, where $\bar{y}$ is the output value of the fuzzy system, has to be derived using the obtained knowledge base. The defuzzification is considered completed if a specific value for each linguistic variable is obtained. To accomplish that, the activity degree of the k-th rule is calculated using formula $\tau^{(k)} = \mu_{A_1^{(k)}}(x_1) \times \mu_{A_2^{(k)}}(x_2)$. Actually, it is determined which of the obtained rules is more active for the specified input data vector.

Rule R^1 from the above-mentioned example has the activity degree $\tau^{(1)} = \mu_{L_1}(x_1) \times \mu_{M_2}(x_2)$. Now, a method of gravity centre determination, say, defuzzification by the gravity centre method *(COGS - Centre of Gravity for Singleton)* [20] can be employed to calculate the output value $\bar{y}$:

$$\bar{y} = \frac{\sum_{k=1}^{N} \tau^{(k)} \bar{y}^{(k)}}{\sum_{k=1}^{N} \tau^{(k)}} \tag{7.7}$$

After all the five stages have been completed successfully, it can be considered that a fuzzy rule base is generated.

Carrying out the calculation for obtaining the rules with data sample given in Table 7.2 and using the above mentioned method, three rules were obtained:

Rule 1: If X1 is MF1 to degree 1.0 and X2 is MF2 to degree 1.0 Then Class is 2 to degree 0.95;

Rule 2: If X1 is MF1 to degree 0.8 and X2 is MF1 to degree 0.6 Then Class is 2 to degree 0.40;

Rule 3: If X1 is MF2 to degree 1.0 and X2 is MF1 to degree 1.0 Then Class is 1 to degree 0.90.

After the rules are obtained, it is necessary to check their "quality". Since the rules obtained contain linguistic variables, as a result of the defuzzification process we will rewrite activities of all rules (see Table 7.4).

Now let us accomplish rule check procedure using Eq.(7.7). As an example, the first input data vector (0.14; 0.85) is considered.

Table 7.4: Input data and activities of the rules obtained.

X_1	X_2	Rule 1		Rule 2		Rule 3		Classes	
0.14	0.85	1	1	1	0	0	0	0	1
0.28	0.42	0.8	0.4	0.8	0.6	0.2	0.6	0	1
0.42	0.71	0.6	0.8	0.6	0.2	0.4	0.2	0	1
0.57	0.28	0.4	0.2	0.4	0.8	0.6	0.8	1	0
0.71	0.57	0.2	0.6	0.2	0.4	0.8	0.4	1	0
0.85	0.14	0	0	0	1	1	1	1	0

Table 7.5: Rules derived from the IRIS database.

Rule 1:	if X1 is MF1 to degree 0.75 and X2 is MF2 to degree 0.87 and X3 is MF1 to degree 0.92 and X4 is MF1 to degree 1.00 then Class is 1 to degree 0.57
Rule 2:	if X1 is MF1 to degree 0.94 and X2 is MF1 to degree 0.88 and X3 is MF1 to degree 0.95 and X4 is MF1 to degree 0.92 then Class is 1 to degree 0.68
Rule 3:	if X1 is MF2 to degree 0.67 and X2 is MF1 to degree 0.58 and X3 is MF2 to degree 0.68 and X4 is MF2 to degree 0.67 then Class is 2 to degree 0.17
Rule 4:	if X1 is MF1 to degree 0.67 and X2 is MF1 to degree 0.75 and X3 is MF2 to degree 0.58 and X4 is MF1 to degree 0.54 then Class is 2 to degree 0.15
Rule 5:	if X1 is MF2 to degree 0.56 and X2 is MF2 to degree 0.54 and X3 is MF2 to degree 0.63 and X4 is MF2 to degree 0.63 then Class is 2 to degree 0.11
Rule 6:	if X1 is MF1 to degree 0.80 and X2 is MF1 to degree 1.00 and X3 is MF1 to degree 0.58 and X4 is MF1 to degree 0.63 then Class is 2 to degree 0.28
Rule 7:	if X1 is MF2 to degree 0.56 and X2 is MF1 to degree 0.88 and X3 is MF2 to degree 0.58 and X4 is MF1 to degree 0.50 then Class is 2 to degree 0.13
Rule 8:	if X1 is MF1 to degree 0.75 and X2 is MF1 to degree 0.71 and X3 is MF1 to degree 0.51 and X4 is MF2 to degree 0.54 then Class is 2 to degree 0.14
Rule 9:	if X1 is MF1 to degree 0.53 and X2 is MF1 to degree 0.71 and X3 is MF2 to degree 0.69 and X4 is MF2 to degree 0.63 then Class is 2 to degree 0.15
Rule 10:	if X1 is MF1 to degree 0.53 and X2 is MF2 to degree 0.58 and X3 is MF2 to degree 0.59 and X4 is MF2 to degree 0.63 then Class is 2 to degree 0.11
Rule 11:	if X1 is MF2 to degree 0.94 and X2 is MF2 to degree 0.75 and X3 is MF2 to degree 0.97 and X4 is MF2 to degree 0.88 then Class is 3 to degree 0.57
Rule 12:	if X1 is MF1 to degree 0.83 and X2 is MF1 to degree 0.79 and X3 is MF2 to degree 0.59 and X4 is MF2 to degree 0.67 then Class is 3 to degree 0.25
Rule 13:	if X1 is MF2 to degree 0.94 and X2 is MF1 to degree 0.75 and X3 is MF2 to degree 1.00 and X4 is MF2 to degree 0.92 then Class is 3 to degree 0.62

Activity of Rule 1: $(0.14\times1+0.85\times1)\ /\ (1+1)=0.495$
Activity of Rule 2: $(0.14\times1+0.85\times0)\ /\ (1+0)=0.14$
Activity of Rule 3: $(0.14\times0+0.85\times0)\ /(0+0)=0$.

Thus we come to a conclusion that data vector (0.14; 0.85) corresponds to the activities of Rule 1 and Rule 2, which ascribe that vector to Cluster 2, as the obtained rules foresee that.

7.6 Example: Iris data set

A well-known IRIS database [4] was selected to perform experiments. The objective of the experiments was:

1. To acquire rules from IRIS data base using the FCM algorithm.
2. To ascertain the effect of membership function number on the count of acquired rules.
3. To check the quality of the rules obtained.

In the first part of the experiments, 3 membership functions were calculated for 3 clusters. Four rules were acquired for Class 1, three rules for Class 2 and 11 rules for Class 3. The rules are shown in Table 7.5 [7].

In the second part of experiments, different initial values of membership functions were selected and rule obtaining was performed (see Table 7.6).

Table 7.6: Dependence of the obtained rule number on the count of initially set membership functions.

Number of memberships	Class 1	Class 2	Class 3	Count
2	2	8	3	**13**
3	4	3	11	**18**
4	7	11	13	**31**
5	13	16	18	**47**
6	21	19	22	**62**

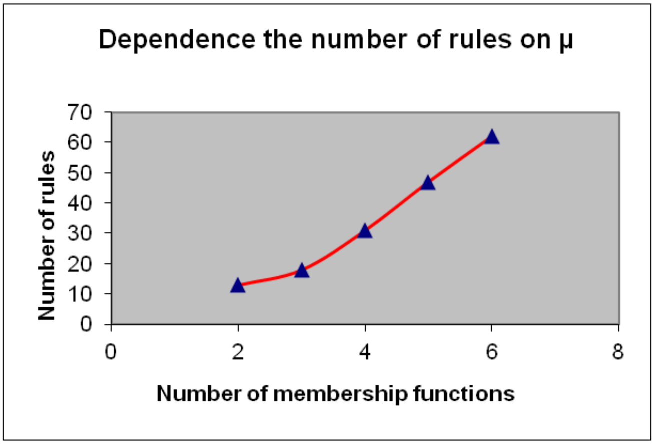


Figure 7.13: The graph of dependence of the number of rules on the number of membership functions.

Dependence shown graphically in Fig.(7.13).

The third part of experiments checks the quality of the obtained rules (see Table 7.5). As a result of the experiments performed, it was stated that the rules obtained for the particular class correctly describe class elements, i.e. for any data vector at least one obtained rule will suit, which brings that data vector to the corresponding class. It should be noted that by increasing the number of functions (as shown in Table 7.6), it is possible to obtain sufficiently many rules, which describe the data accurately.

7.7 Conclusions

This chapter presents the results of the research on fuzzy rule construction. The obtained rules form a rule base of particular data and are represented in the form of *IF-THEN* rules. Methods of fuzzy rule base design are mostly used in fuzzy control systems. This chapter, however, concentrates on the description of the methodology that enables obtaining fuzzy rules from the numerical data. The major advantage of that methodology is a possibility of rule interpretation in the form clear to the user. Since the method employs elements of fuzzy set theory, the main problem is related to the clustering of initial data that allows one to determine object allocation to classes and obtain characteristics of membership functions. To achieve that, the fuzzy clustering algorithm FCM is used. After the membership functions are determined, the procedure of designing a base of fuzzy rules is performed in 5 stages: (1) dividing the space of input and output signals into areas, (2) constructing fuzzy rules on the basis of learning data, (3) ascribing truth degree to each rule, (4) creating a base of fuzzy rules, and finally (5) defuzzification intended to calculate output values of the fuzzy system. The stages are illustrated with an example that characterizes the essence of the method. The experimental part contains method implementation on the basis of the famous IRIS database. In the course of experiments, fuzzy rules were obtained using a method of fuzzy rule base design at different values of the count of membership functions. The experiments have shown that the rules obtained correctly describe the initial data. Thus, a base of fuzzy rules was obtained, which enables one to positively evaluate the methodology discussed in the paper under consideration.

References

[1] M.N. Ahmed, S.M. Yamany, N. Mohamed, A.A. Farag, and T. Moriarty. A modified fuzzy c-means algorithm for bias field estimation and segmentation of MRI data. *IEEE Transactions on Medical Imaging*, 21(3):193–199, 2002.

[2] R. Bellman, R. Kalaba, and L. Zadeh. Abstraction and pattern classification. *Journal of Mathematical Analysis and Applications*, 13(1):1–7, 1966.

[3] J.C. Bezdek. *Pattern Recognition with Fuzzy Objective Function Algorithms*. Plenum Press, New York, USA, 1981.

[4] Iris Data Set. The UCI Machine Learning Repository. url-http://archive.ics.uci.edu/ml/datasets/Iris.

[5] R.O. Duda and P.E. Hart. *Pattern Classification and Scene Analysis*. Wiley, New York, USA, 1973.

[6] J.C. Dunn. A fuzzy relative of the ISODATA process and its use in detecting compact well-separated clusters. *Journal of Cybernetics*, 3(3):32–57, 1974.

[7] P.S. Grabusts. Application of fuzzy rule base design method. *Scientific Proceedings of Riga Technical University: Computer Science. Techonologies of computer control, RTU*, 24(5):125–131, 2005.

[8] F. Hoppner, F. Klawonn, R. Kruse, and T. Runkler. *Fuzzy Cluster Analysis: Methods for Classification, Data Analysis and Image Recognition*. Wiley, New York, USA, 1999.

[9] J.S.R. Jang, C.T. Sun, and E. Mizutani. *Neuro-Fuzzy and Soft Computing: A Computational Approach to Learning and Machine Intelligence*. Prentice Hall, New Jersey, USA, 1997.

[10] F. Klawonn and A. Keller. Fuzzy clustering and fuzzy rules. In *7th International Fuzzy Systems Association World Congress (IFSA)*, volume I, pages 193–198, 1997.

[11] G.J. Klir and B. Yuan. *Fuzzy Sets and Fuzzy Logic: Theory and Applications*. Prentice Hall, New Jersey, USA, 1995.

[12] R. Nock and F. Nielsen. On weighting clustering. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 28(8):1223–1235, 2006.

[13] E.H. Ruspini. A new approach to clustering. *Information and Control*, 15(1):22–32, 1969.

[14] T. Takagi and M. Sugeno. Fuzzy identification of systems and its applications to modeling and control. *IEEE Transactions on Systems, Man, and Cybernetics*, SMC-15(1):116–132, 1985.

[15] I.B. Turksen. A review of developments in fuzzy system models: fuzzy rule bases to fuzzy functions. *Scientia Iranica*, 18(3):522–527, 2011.

[16] L.X. Wang. Solving fuzzy relational equations through network training. In *2nd IEEE International Conference on Fuzzy Systems*, volume 2, pages 956–960, 1993.

[17] L.X. Wang and J.M. Mendel. Generating fuzzy rules by learning from examples. *IEEE Transactions on Systems, Man, and Cybernetics*, 22(6):1414–1427, 1992.

[18] M.S. Yang. A survey of fuzzy clustering. *Mathematical and Computer Modelling*, 18(11):1–16, 1993.

[19] M.S. Yang, Y.Z. Chen, and Y. Nataliani. Bias-correction fuzzy c-regressions algorithm. In *14th International Conference on Artificial Intelligence and Soft Computing (ICAISC)*, volume 9119 of *LNAI*, pages 283–293, 2015.

[20] M.S. Yang, C.Y. Lin, and Y.C. Tian. A robust fuzzy classification maximum likelihood clustering framework. *International Journal of Uncertainty, Fuzziness and Knowledge-Based Systems*, 21(5):755–776, 2013.

[21] L.A. Zadeh. Fuzzy sets. *Information and Control*, 8(3):338–353, 1965.