CONSTRUCTION OF FUZZY IF-THEN RULES BY CLUSTERING AND FUZZY DECISION TREE LEARNING

Dinh Manh Tuong
Faculty of Technology, VNU

Abstract. In this paper we propose the method of constructing fuzzy if-then rules from a set of input-output data. This method consists of two steps. We first construct fuzzy sets covering input and output spaces by clustering. Then applying decision tree learning algorithm with some suitable changes we construct the fuzzy decision tree. From this tree we can generate fuzzy if-then rules.

1. INTRODUCTION

A fuzzy system consists of two basic components: the fuzzy rule base and the fuzzy inference engine. Fuzzy systems have been applied in many fields such as control, signal processing, communications, integrated circuit manufacturing, and expert systems to business, medicine, etc. The fuzzy rule base comprises the following fuzzy if-then rules:

$$\text{If } \mathbf{x}_1 \text{ is } A_1 \text{ and ... and } \mathbf{x}_n \text{ is } A_n \text{ then y is B },$$

where A_i are fuzzy sets in input spaces $X_i \subset R$ ($i = 1, ..., n$), and B is a fuzzy set in output space $Y \subset R$. In many application domains, when developing the fuzzy system, by observing we obtain a set of input-output data. There are many methods of designing fuzzy systems from the set of input-output data (see [3, 5, 6]). The design of fuzzy systems from input-output data may be classified in two types of approaches. In the first approach, fuzzy if-then rules are first generated from input-output data, then other components of the fuzzy system are constructed from these rules according to certain choice of fuzzy inference engine, fuzzifier, defuzzifier. In the second approach, the structure of the fuzzy system is specified first with some parameters in the structure, and then these parameters are determined according to the input-output data.

In this paper we propose the method of constructing fuzzy if-then rules from a set of input-output data by clustering and fuzzy decision tree learning. In the section 2 we construct the fuzzy set systems that cover the input and output spaces. In the section 3 the fuzzy decision tree will be constructed.

2. CONSTRUCTION OF COMPLETE AND CONSISTENT SYSTEMS OF FUZZY SETS FOR THE INPUT AND OUTPUT SPACES

Suppose that we need to design a system with n inputs $\mathbf{x}_1, ..., \mathbf{x}_n$ and a output y. Each variable $\mathbf{x}_i$ ($i = 1, ..., n$) obtains values in the space $X_i = [a_i, b_i] \subset R$, and y obtains values in the space $Y = [c, d] \subset R$.

Suppose that we are given the set D of input-output data pairs (x, y), where x = (x_1, ..., x_i, ..., x_n) is the vector of inputs, y is the output according to x. Our objective is to construct fuzzy if-then rules from the set D of input-output data.

We denote:

$D_i = \{x_i \mid x_i$ is the i^{th} component of x, $(x, y) \in D\}$

$D' = \{y \mid (x, y) \in D\}$

Therefore D_i is the set of points in the space $X_i = [a_i, b_i]$ (i = 1, ..., n), and D' is the set of points in the space Y = [c, d]. We want to construct fuzzy sets A^i_1, ..., A^i_{mi} that cover the space X_i from the data set D_i (i = 1, ..., n), and construct fuzzy sets B_1, ..., B_m that cover the space Y from the data set D'.

Definition:

1. The system of fuzzy sets A_1, ..., A_m in the space X is called a complete system if for any $x \in X$ there exists a fuzzy set A_k ($1 \le k \le m$) such that the membership degree of x in the fuzzy set A_k is greater than zero, that is $\mu_{Ak}(x) > 0$.

2. The system of fuzzy sets A_1, ..., A_m is called a consistent system if at arbitrary x $\in$ X that $\mu_{Ak}(x) = 1$ then $\mu_{Aj}(x) = 0$ for all $j \neq k$, $1 \le j \le m$.

The complete and consistent system of fuzzy sets in the space X is called a cover of space X.

Suppose that D is a set of points in X = [a, b]. From the data set D we construct fuzzy sets A_1, ..., A_m that cover X as follows. We first divide the data set D into m clusters C_1, ..., C_m using clustering algorithms. For doing that we can apply one of following well-known clustering algorithms: k-Means, k-Medoids, or DBSCAN (see [1, 2]). Suppose that x_0 is the center of cluster C, then the radius r of cluster C is defined as

$$r = \frac{1}{|C|} \sum_{x \in C} |x - x_o|$$

From the centers and radii of clusters C_1, ..., C_m we have alternatives to construct the fuzzy sets A_1, ..., A_m covering the space X = [a, b]. For example, if the data set D is grouped into three clusters with centers x_i (i = 1, 2, 3) and radii r_i (i = 1, 2, 3) corresponding, then we can construct trapeziform fuzzy sets A_1, A_2, A_3 as specified in the figure 1.

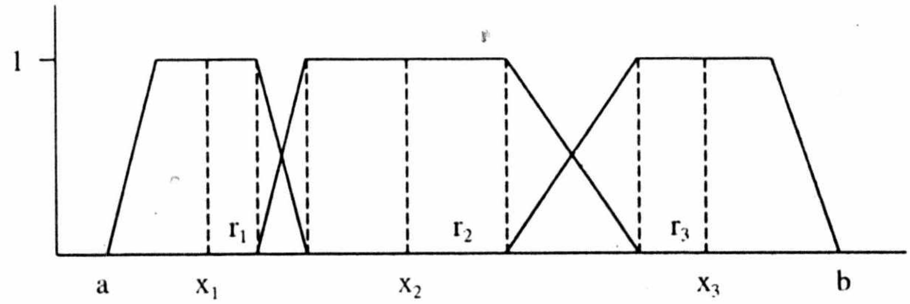

Figure 1. Trapeziform fuzzy sets covering X= [a, b]

Using the above represented method, by clustering the data set D_i into the cluster $C^i_1, \ldots, C^i_{mi}$, we construct the fuzzy sets $A^i_1, \ldots, A^i_{mi}$ covering the input space $X_i = [a_i, b_i]$ $(i = 1, \ldots, n)$. Analogically, from the data set D' we construct the fuzzy sets $B_1, \ldots, B_m$ covering the output space $Y = [c, d]$.

For $x_i \in X_i$, we say that "x_i is A^i_j" if j is defined as:

$$j = \arg \max_{1 \le k \le m_i} \mu_{A^i_k}(x_i)$$

Analogically, we say that "y is B_t" if t is defined as:

$$t = \arg \max_{1 \le j \le m} \mu_{B_j}(y)$$

That is, B_t is a fuzzy set such that $\mu_{B_t}(y) = \max\{\mu_{B_1}(y), \ldots, \mu_{B_m}(y)\}$

By the above represented technique, we have discretizated the continuous space $X_i = [a_i, b_i]$ by finite number of values $A^i_1, \ldots, A^i_{mi}$. Each i^{th} input $\mathbf{x}_i$ will obtain one of the fuzzy sets $A^i_1, \ldots, A^i_{mi}$ as his value. Analogically, the output **y** can obtain one of fuzzy sets $B_1, \ldots, B_m$ as his value.

3. CONSTRUCTION OF THE FUZZY DECISION TREE

In this section we present the method of constructing the fuzzy decision tree from the set D of input-output data. The fuzzy decision tree will be constructed by applying the decision tree learning algorithm (see [2, 4]) with some suitable changes.

In the fuzzy decision tree when the input $\mathbf{x}_i$ is the label of a node, below this node there are m_i branches with labels being fuzzy sets $A^i_1, \ldots, A^i_{mi}$ as in the figure 2.

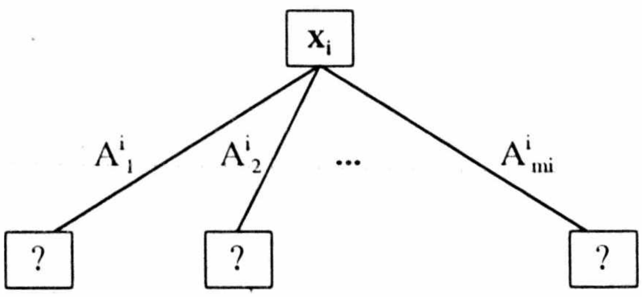

Figure 2. A node of the fuzzy decision tree with label x_i

The fuzzy decision tree will be constructed by developing the tree starting from the single-node tree. In each step, an unlabeled leaf node will be selected to develop (to develop a node means to assign a label to this node and to define branches (if any) going down from this node). In the process of developing the tree, when a node is selected to develop (a node that is selected to develop is called ***current node***), we should choose an input variable to be the label of this node. If the variable $\mathbf{x}_i$ is selected to be the label of current node, there will be m_i branches with labels $A^i_1, ..., A^i_{mi}$ going down from this node as in the figure 2. The current node can be not developed and will become a leaf of resulting decision tree. This leaf will be labeled by one of the fuzzy sets $B_1, ... B_m$. The choice of label of the current node is performed by using the entropy measure. The entropy is employed as a measure of the impurity in a collection of data.

Given a subset S of the set D of input-output data

$$S = \{(x, y) \mid (x, y) \in D\}$$

Because the output can take on m different values as fuzzy sets $B_1, ..., B_m$, the entropy of S is defined as follows

$$\alpha_j = \sum_{(x,y)\in S} \mu_{B_j}(y) \tag{1}$$

$$\alpha = \sum_{j=1}^{m} \alpha_j$$

$$p_j = \frac{\alpha_j}{\alpha} \qquad (j = 1, ..., m)$$

$$Entropy(S) = \sum_{j=1}^{m} - p_j \log_2 p_j \tag{2}$$

Suppose that S is the data set going into the node with label $\mathbf{x}_i$ (see the figure 2), we denote S_j as the subset of S, S_j consists of all data going into the branch with label A^i_j. The expected entropy of S after using the variable $\mathbf{x}_i$ to partition S, denoted by ExpEntropy(S,$\mathbf{x}_i$), is defined as follows

$$ExpEntropy(S, x_i) = \sum_{j=1}^{m} \frac{s_j}{s} Entropy(S_j) \tag{3}$$

where s_j is the number of elements of S_j, $s_j = |S_j|$, and $s = |S|$. Therefore, the expected entropy is simply the sum of the entropies of each subset S_j, weighted by the fraction of data that belong to S_j.

We now assume that a node is selected to develop and S is the set of data going into this node. The input variable $\mathbf{x}_i$ will be assigned to be the label of current node if ExpEntropy (S, $\mathbf{x}_i$) is the smallest in all ExpEntropy(S, $\mathbf{x}_j$), where $\mathbf{x}_j$ is not on the road from the tree root to the current node. That is, if we denote IND as the set of indices j such that $\mathbf{x}_j$ are not label of nodes on the road from the tree root to the current node, then $\mathbf{x}_i$ will be the label of the node, where i is defined by

$$i = \arg \min_{j \in IND} ExpEntropy\,(S, x_j) \qquad (4)$$

If the set of data S goes into a branch and Entropy(S) is small enough, that is Entropy(S)< ε, where ε is a given positive constant, then this branch will go into a leaf of resulting decision tree, this leaf is labeled by B_k, where k is defined as

$$k = \arg \max_{1 \le j \le m} \alpha_j \qquad (5)$$

The following is algorithm of constructing the fuzzy decision tree.

Algorithm:

- Create a root node of the tree with the data set D going into this mode

- Repeat

1. Select a node (an unlabeled leaf of the current tree) to develop. Suppose that S is the data set going into the current node.

2. If S is empty then the current node is a leaf of the resulting tree, this leaf is labeled by B_k, where k is defined by (5) with

$$\alpha_j = \sum_{(x,y) \in S'} \mu_{B_j}(y)$$

where S' is the data set going into the parent node of the current node

3. Else Begin

3.1. If Entropy(S) < ε or IND is empty then the current node is a leaf of the resulting tree with label B_k, where k is defined by (1) and (5).

3.2. Else Begin

3.2.1. The current node is labeled by $\mathbf{x}_i$ where i is defined by (4).

3.2.2. Below this node there are m_i branches with labels A^i_1, ..., A^i_{mi}, these branches lead to new nodes. The data set going into the branch A^i_j is S_j (j = 1, ..., m_i), where S_j is defined as

$$S_j = \left\{ (x, y) \in S \;\middle|\; \arg \max_{1 \le k \le m_i} \mu_{A^i_k}(x_i) = j \right\}$$

End

End

Until all leaves of the tree are labeled.

From the constructed fuzzy decision tree, we can construct fuzzy if-then rules. Each road from the tree root leading to a leaf generates a fuzzy if-then rule.

CONCLUSION

Above we proposed the method of constructing fuzzy if-then rules from a set of input-output data. This method consists of two steps. We first construct the systems of fuzzy sets covering the input and output spaces by using well-known clustering techniques. Then applying decision tree learning algorithm with some suitable changes we construct the fuzzy decision tree. From this tree we generate fuzzy if-then rules, each rule is corresponding to a road from the tree root leading to a leaf.

REFERENCES

1. M. Ester, H. P. Kriegel, J. Sander, and X. Xu. A density-based algorithm for discovering clusters in large spatial databases, *Proceedings of 2nd Int. Conf. on Knowledge Discovery and Data Mining*, Portland, OR, Aug., 1996, pp.226-231.
2. J. Han and M. Kamber. *Data Mining: Concepts and Techniques*, Morgan Kaufmann, Publishers, San Francisco, 2001, 550p.
3. C. T. Lin. *Neural fuzzy control systems with structure and Parameter Learning*, World Scientific, Singapore, 1994, 318p.
4. T, M. Mitchell. *Machine Learning*, McGraw-Hill, Inc., 1997, 414p.
5. L. X. Wang and J. M. Mendel. Generating fuzzy rules by learning from examples, *IEEE Trans. On Systems, Man, and Cybern.*, 22(6), 1992, pp.161-170.
6. L. X. Wang. *A course in Fuzzy Systems and Control*, Prentice-Hall Int., Inc., 1997, 424p.

TẠP CHÍ KHOA HỌC ĐHQGHN, KHTN & CN, T.XX, Số 2, 2004

XÂY DỰNG CÁC LUẬT IF – THEN MỜ BẰNG PHƯƠNG PHÁP PHÂN CỤM VÀ HỌC CÂY QUYẾT ĐỊNH MỜ

Đinh Mạnh Tường
Khoa Công nghệ, ĐHQG Hà Nội

Trong bài này chúng tôi đề xuất một phương pháp xây dựng các luật if-then mờ từ một tập các cặp dữ liệu vào-ra. Phương pháp này gồm hai giai đoạn. Đầu tiên áp dụng kỹ thuật phân cụm, chúng ta xây dựng các hệ tập mờ phủ các không gian dữ liệu vào-ra. Sau đó bằng cách áp dụng thuật toán học cây quyết định với một số thay đổi thích hợp, chúng ta xây dựng cây quyết định mờ. Các luật if-then mờ sẽ được hình thành từ cây quyết định này.