An Algorithm for Rule Generation in Fuzzy Expert Systems

Kropotov Dmitry
dkropotov@yandex.ru

Vetrov Dmitry
vetrovd@yandex.ru

Dorodnicyn Computing Centre of the Russian Academy of Sciences
Russia, 119991, Moscow, GSP-1,
Vavilova str. 40

Abstract

Although using fuzzy logic in control systems has become widely established as an appropriate approach, its application in area of pattern recognition and data mining seems to be restricted. These systems have several bottlenecks mainly concerning fuzzy rules generation and fuzzy sets forming. The state-of-the-art technics here is neuro-fuzzy approach which has some disadvantages. In the presented article there considered an algorithm for rules generation based on alternative principles and some ideas on defining fuzzy sets.

1 Introduction

At the present time fuzzy logic concept finds its application in many areas of human knowledge. Thus there exist a lot of successive projects of implementing fuzzy logic in control systems [7]. The ability of the theory to represent dependencies in linguistic terms facilitating understanding and managing the investigated process [8] led to development of fuzzy expert systems [5]. Such systems aimed for supervized learning or forecasting fall under the situation, in which we are given a set of fuzzy sets for each feature and knowledge base - a set of fuzzy rules. The successive system's creation depends fully on happy choice of fuzzy sets and rules appropriate for the current research field. It is a common situation than experts can't properly solve the problem with forming of fuzzy sets and rules and hence there is a need of providing some kind of automatic means. The most of known software products intended for fuzzy expert system's development (MatLab, CubiCalc, etc.) uses so-called neuro-fuzzy approach based on attraction of neural networks for rule generation and some heuristics of uniform partitioning for assigning of fuzzy sets [2],[1]. The neuro-fuzzy approach has some sufficient drawbacks:

- The fuzzy system with great number of generated rules with relatively low significance level tends to sufficient overfitting
- High rules dimensions lead to poor knowledge interpretation and inability of deep understanding for the current application field
- Great time of calculation taken from corresponding problem of neural networks

The goal of this article is to establish rules generating algorithm which avoids the mentioned drawbacks and at the same time provides an appropriate knowledge base. In the next section such algorithm is considered in details. Section 3 gives some heuristic for fuzzy sets assigning. In section 4 we demonstrate the practicality of proposed approach using experimental results and conclude with some discussion.

2 Rule generation

Consider the following decision-making task. There are n parameters of the system $\vec{X} = (X_1 \dots X_n) \in \mathbb{R}^n$, which can be measured and one variable $Y \in \mathbb{R}$ to be predicted. Our goal is to get a set of fuzzy rules appropriate for our task with the aid of learning sample $\{\vec{x}^k, y^k\}_{k=1}^p$. Fuzzy rules will be formed in following general form:

$$R: \ IF \ K_1 \vee K_2 \vee \ldots \vee K_q \ THEN \ Y \in B_k. \quad (1)$$

where K_i is denotation for some conjunction:

$$K \triangleq X_{i_1} \in A_{k_1}^{i_1} \& \ldots \& X_{i_l} \in A_{k_l}^{i_l} \quad (2)$$

Here X_{i_p} is a value of i_p-th feature and $A_{k_p}^{i_p}$ is k_p-th fuzzy set of i_p-th feature. Let's denote for some rule

R a set of conjunctions $\{K_1, \dots, K_q\}$ as a $Sump(R)$ – sumption of the rule and B_k as a $Res(R)$ – result set of the rule. $\mu_A(x) : X \to [0,1]$ designates a membership function of fuzzy set A. Then membership function of conjunction (2) will be $\mu_K(\vec{x}) = \min(\mu_{A_{k_1}^{i_1}}(x_{i_1}^j), \dots, \mu_{A_{k_l}^{i_l}}(x_{i_l}^j))$.

Definition 1. Representativeness *of rule is the following value:*

$$rep(R) \triangleq \frac{1}{p} \sum_{j=1}^{p} \max(\mu_{K_1}(\vec{x}^j), \dots, \mu_{K_q}(\vec{x}^j))$$

Definition 2. Effectiveness *of rule is given by the following formula:*

$$eff(R) \triangleq \frac{\sum_{j=1}^{p} \min(\max(\mu_{K_1}(\vec{x}^j), \dots, \mu_{K_q}(\vec{x}^j)), \mu_{B_k}(y^j))}{p \cdot rep(R)}$$

In other words, representativeness is implicitly the rate of precedents, which satisfy the sumption of the given rule while effectiveness is implicitly the rate of precedents from the sumption, which satisfy the rule itself. We intend to generate rules, which have both high representativeness and effectiveness.

The algorithm consists of two stages. On the first stage we try to generate rules only with one conjunction in their sumption while all other rules are generated on the second stage.

2.1 Generation of conjunctive rules

Consider only the rules with one conjunction in their sumptions. In this section we call the number of fuzzy sets in conjunction (i.e. sumption) as **order of the rule**.

Definition 3. Rule R_b is **restriction** of rule R_a if the next two conditions are satisfied:

- $Res(R_a) = Res(R_b)$
- The sumption of rule R_b contains all fuzzy sets from the sumption of rule R_a

During the rule restriction representativeness becomes lower while the effectiveness may become higher. In the last case we will call restriction an **effective** one.

Set c_1 and c_2 - thresholds of representativeness and effectiveness correspondingly. A rule is **tolerable** if its effectiveness is more than c_2 and representativeness is more than c_1. An algorithm given below (effective restrictions method) allows finding all tolerable rules of minimal possible order according to learning sample. It is based on linear search over the rules order.

1. Fix B_k - result set of the rule.

2. Construct all possible rules of first order from the fuzzy sets we have, i.e. rules of type $IF\ x_{i_1} \in A_{k_1}^{i_1},\ THEN\ y \in B_k$.

3. Reject all rules that have $rep(R) < c_1$.

4. If no rules remained then go to step 6. Otherwise examine the effectiveness of residuary rules. If $eff(R) > c_2$ then the rule is tolerable and should be moved to the list of final rules.

5. All other rules (if any) are used for restrictions in a following way. Sumption of any rule being restricted should be a subset of any other two rules, which are being restricted to the same rule of higher order. In other words, the union of sumptions of any two rules, which are restricted to the same rule of higher order, is exactly the sumption of this new rule (see Figure 1). If no new rules got, then go to step 6. Otherwise go to step 3.

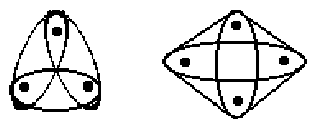

Figure 1. Restriction of rules to third and forth order. *Points represent fuzzy sets and contours encircle rules sumptions.*

6. If all result sets were examined then stop working, otherwise increase k by one and go to step 1.

Consider the selection of c_1 and c_2 thresholds in details. The aim is to find all significant rules without overtraining. In other words the rate of noise rules should be low enough in order the rule system to be adequate. It is clear that the effectiveness threshold $c_2 = c_2(rep(R))$ should be higher with the decrease of representativeness value. Let we have crisp (i.e. non-fuzzy) sets. The rule is insignificant if the information that object satisfies the rule sheds no light on its affiliation to the result set of the rule. Let's check the following statistical hypothesis:

$$\mathbb{P}\{y^i \in Res(R) | \vec{x^i} \in Sump(R)\} = \mathbb{P}\{y^i \in Res(R)\}$$

Without loss of generality suppose uniform prior probabilities, i.e. $\mathbb{P}\{y^i \in B_1\} = \ldots = \mathbb{P}\{y^i \in B_r\} = 1/r$. Examine the value $rep(R)eff(R)p$. If the hypothesis is right, we have $n = rep(R)p$ Bernoulli trials with the probability of success equals $s = 1/r$. If $ns > 5$

(this can be adjusted by setting $c_1 > 5r/p$), according to Moivre-Laplace theorem, the distribution can be approximated with a normal distribution ([6],[3]) with the mean of ns and variance of $\sigma^2 = ns(1-s)$. This means that:

$$eff(R) \sim N(s, s(1-s)/n)$$

Fixing the level of significance α, we find the necessary effectiveness threshold

$$c_2 = \frac{1}{r} + \frac{z_\alpha\sqrt{r-1}}{r\sqrt{rep(R)p}} \qquad (3)$$

where z_α is fractile of standard normal distribution.

Now let's exclude on each iteration from further consideration the rules, which will not become tolerable even under the most favourable conditions. These are rules with so low effectiveness value which cannot be made big enough after any restrictions. Suppose that during restrictions the only objects to exclude are those, which do not belong to $Res(R)$ (the most favourable case). The effectiveness of the rule will increase in the fastest way with such restrictions. The rule will be intolerable if its effectiveness is less than c_2 when its representation equals c_1. Now it is easy to get conditions of a fortiori intolerable rules:

$$eff(R) < \frac{c_1}{rep(R)r} + \frac{z_\alpha\sqrt{(r-1)c_1}}{\sqrt{p}rep(R)r} \qquad (4)$$

All other restrictions will lead to even lesser effectiveness value of the rule. The last condition allows to reduce the time of rule generation process significantly, by exclusion of large number of rules without loss of tolerable ones. So now with the aid of formulas (3),(4) restriction process goes as shown on Figure 2.

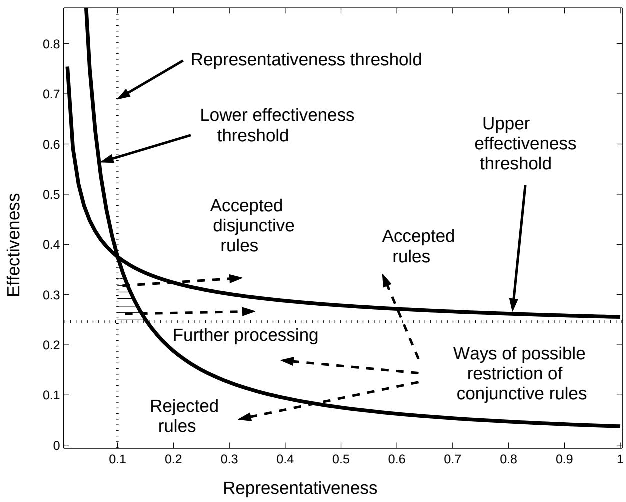

Figure 2. Restriction procedure. *Each rule can be represented as a point in effectiveness-representativeness space.*

2.2 Generation of disjunctive rules.

Consider the general case of rules of form (1). During the disjunctive restriction (i.e. conjunction merging) representativeness becomes higher while effectiveness stays between corresponding characteristics of rules being restricted. Consequently disjunctive restriction of arbitrary rules may easily lead to tolerable dependency. Therefore there is a need for some limitation of the class of rules for disjunctive restriction. In this section **the order of the rule** is understood as number of conjunctions included in rule's sumption. The proposed algorithm searches as before over rules orders.

1. Take the rules with one conjunction in their sumptions which are enough representative and effective (i.e. their $rep(R) > c_1, eff(R) >= c_2(1)$), but don't have opportunities to become tolerable during previous restriction (see Figure 2)
2. The rules being restricted to rules of second order must have some interconnections in their sumptions:
   (a) For rules with only one fuzzy set in their sumptions the used feature must coincide
   (b) For all others at least one fuzzy set in one feature must coincide
3. The rules restriction goes in similar manner as in 2.1. Sumption of any rule being restricted should be a subset (in a sense of conjunctions sets) of any other two rules, which are being restricted to the same rule of higher order.
4. If no rules remained then stop. Otherwise examine the effectiveness of residuary rules. If $eff(R) > c_2$ then the rule is tolerable and should be moved to the list of final rules. Go to step 3.

3 Defining fuzzy sets.

As it was mentioned above assigning the shapes and locations of fuzzy sets for expert may be quite a difficult problem. At the same time, expert can relatively easy indicate the approximate borders of fuzzy sets [5]. According to this fact, suppose the idea of parametrical family of fuzzy set's shapes establishing. In [4] using only isosceles triangle and trapezium as forms of fuzzy sets leads to parametrical family with only $2n$ parameters to be optimized preserving the great variety of possible shapes .

4 Results and conclusions

The concepts described above were implemented in the program ExSys, which can be used both for pattern recognition and forecasting tasks depending on selected defuzzification method. In recognition case its results were compared with q-nearest neighbors (QNN), support vector machines (SVM), committee of linear classificators (LM), test algorithm (TA), linear Fisher discriminant (LDF) and multiple layer perceptron (MLP). The comparison was help according to three applications. The first was melanoma diagnostics (3 classes, 33 features, 48 objects in the training sample), second was speech phoneme recognition (2 classes, 5 features, 2200 objects in the training sample) and the last was drug intoxication diagnostics (2 classes, 18 features, 450 objects in the training sample). The results of experiments (percent of correctly recognized objects from the independent test sample) are shown on Figure 3.

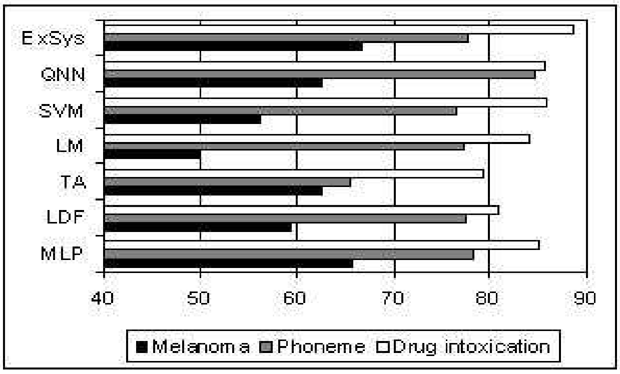

Figure 3. The performance of different recognition algorithms on three real-life tasks.

In area of forecasting ExSys was compared with multiple linear regression and MatLab. There was considered the following task: predictions of magnetic amplitude oscillations in accelerating cavity of a klystron. The necessary data was taken from Hamburg linear accelerator in DESY. The source information was oscillations on other cavities within the same klystron. The same table was used for learning of both systems. The results of their work on the control sample are shown on Figure 4.

The tests show that the methods, described above can be successfully used for fuzzy expert systems development. The proposed algorithm for knowledge base

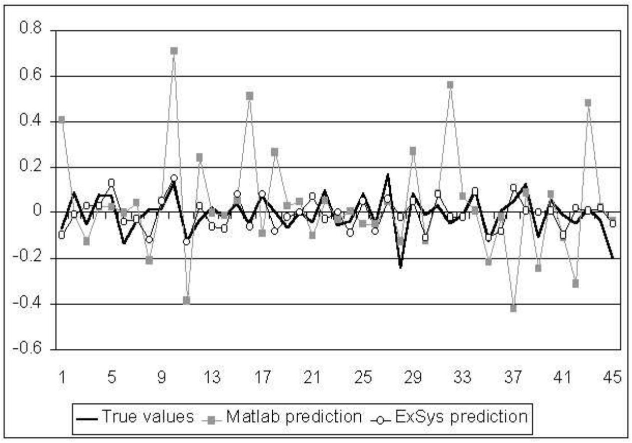

Figure 4. Oscillations of magnetic field amplitude.

generation provides not a great number of rules which are both statistically significant and easily interpreted by experts. The approach focuses on the essence of research problem, not on particular samples, thus preventing the whole system from catastrophic overtraining.

5. Acknowledgements

The work was partially supported by the Russian Foundation for Basic Research (grants 02-07-90134, 02-07-90137, 02-01-00558, 02-01-08007).

References

[1] http://www.hyperlogic.com/cbc.html.

[2] http://www.mathworks.com/access/helpdesk/help/toolbox/fuzzy/fuzzybeg.shtml.

[3] W. Feller. *An introduction to probability theory and its applications*, volume 2. John Wiley & Sons, New York, 1966.

[4] D. Kropotov and D. Vetrov. Using precedent information in fuzzy expert systems. In *Proc. 6th Int. Conf. on Pattern Recognition and Image Analisys: New Information Technologies*, volume 1, pages 100–104, Velikiy Novgorod, Russian Federation, 2002.

[5] I. Perfil'eva. Applications of fuzzy sets theory. In *Itogi nauki i tehniki*, volume 29 of *Teoriya veroyatnostei. Mat. stat. Teor. kibernet.*, pages 83–151, Moscow, 1990. VINITI. (in russian).

[6] L. Sachs. *Statistische Auswertungsmethoden.* Springer-Verlag, 1972.

[7] T. Terano, K. Asai, M. Sugeno, and editors. *Applied Fuzzy Systems.* Number ISBN 0-12-685242-1. Translated by C. Asschmann, AP Professional.

[8] L. Zadeh. *The Concept of a linguistic variable and its application to approximate reasoning*. Elsevier Pub. Co., New York, 1973.