Normalization Issues in Granular Evidence-Based Adaptive Network ANGIE

A. Valishevsky[1] and A. Borisov[2]
[1]Institute of Mathematics and Computer Science, University of Latvia,
Vejavas 10/1-33, LV-1035, Riga, Latvia
E-mail: valisevskis@inbox.lv
[2]Decision Support Systems Group, Technical University of Riga, Kalku 1, LV-1658, Riga, Latvia
E-mail: aborisov@egle.cs.rtu.lv

Abstract

Granular evidence-based adaptive network ANGIE (adaptive network for granular information processing) is considered in this paper. The network described can be used in fuzzy-evidence-based decision support systems. The architecture and learning algorithm underlying ANGIE is presented. The proposed learning procedure helps in solving a task that can be related to sensitivity analysis in decision aid models. However, normalization procedure should be applied to network parameters to avoid inconsistency. This paper considers different approaches towards normalization. The effectiveness and accuracy of different approaches are compared.

Keywords: *adaptive network, information granularity, fuzzy evidence, decision support systems, sensitivity analysis.*

1 Introduction

One of the first attempts to combine neural networks and fuzzy logic and to bring adaptability to the fuzzy systems was made in [1]. The idea of that paper was to use adaptive network as a fuzzy inference system. Learning algorithm was also proposed, that allowed tuning membership functions parameters. The need for adaptive fuzzy systems is evident, as fuzzy-rule-based systems can rarely be applied to solving real world problems without prior tuning of rules.

This paper proposes adaptive network ANGIE (Adaptive Network for Granular Information and Evidence processing) that can be used in fuzzy-evidence-based decision support systems. The learning algorithm is presented as well. However, the algorithm proposed is intended for determining the contribution of fuzzy features which form the left-hand part of fuzzy rules to achieving the desired value of the fuzzy criterion which is indicated in the right-hand part of fuzzy rules. It has a strong relation to sensitivity analysis in the theory of decision-making [2], though it is not quite the same. Given the desired value of the criterion, with the aid of the proposed algorithm it is possible to determine the *importance* of each fuzzy feature for the criterion to take the desired value. The algorithm will be described later in the paper.

The algorithm is based on gradient descent method. Thus, normalization procedure should be applied to network parameters, as they represent probabilities of different events and their consistency should be preserved. There are several possibilities to carry out normalization of system parameters. One can normalize parameters after each iteration, after each *n*-th iteration or after the training process has finished. These approaches are considered in the corresponding chapter.

In the next chapter the basic idea behind information granularity is given. Later on, the adaptive network ANGIE and the training algorithm are described.

2 Information Granularity

The idea of information granularity and its application in the context of fuzzy logic is presented in [3]. The idea of information granularity can be summarized as follows.

Information is contained in granules. In this paper conditioned π-granules are considered which are characterized by propositions of the following form:

$$g_i \stackrel{\Delta}{=} \text{If } X = u_i \text{ then } Y \text{ is } G_i ,$$

where G_i is a fuzzy subset of V which is dependent on u_i. Evidence E can be regarded as a collection of these granules:

$$E = \{g_1, \ldots, g_N\}.$$

Variable X assumes its value with a specified probability. Thus, evidence is probability distribution P_X of conditional π- granules. Moreover, each granule can be regarded as conditional possibility $\Pi_{(Y|X=u_i)} = G_i$. Hence, evidence can be regarded as conditional possibility distribution $\Pi_{(Y|X)}$. Thus, evidence can be considered as the following construction:

$$E = \{P_X, \Pi_{(Y|X)}\}.$$

Given a collection of bodies of evidence $E = \{E_1, \ldots, E_K\}$, one would like to ask some questions about the information contained in these bodies of evidence. The main question is: "what is the probability of (Y is Q)?", where Q is a fuzzy subset of V. The probability of such an event is an interval value in which the expected possibility $\mathrm{E}\Pi(Q)$ is regarded as the upper boundary, and the expected certainty $\mathrm{EC}(Q)$ is regarded as the lower boundary of the probability sought. In [3] the following formulae are proposed:

$$\mathrm{E}\Pi(Q) = \sum_{i,j} p_{ij} \sup\left(Q \cap G_{1,i} \cap G_{2,j}\right), \tag{1}$$

$$\mathrm{EC}(Q) = 1 - \mathrm{E}\Pi(Q'), \tag{2}$$

where Q' is a complement of set Q. Formula (1) is for a special case in which there are two bodies of evidence in the system but it would not present any difficulty to induce a more general form of the formula.

3 Architecture of the ANGIE Adaptive Network

Let us consider a set that consists of two bodies of evidence:

I. Evidence 1:
IF (X_1 is B_{11}) with p_{11} THEN (Y is G_{11})
IF (X_1 is B_{12}) with p_{12} THEN (Y is G_{12})

II. Evidence 2:
IF (X_2 is B_{21}) with p_{21} THEN (Y is G_{21})
IF (X_2 is B_{22}) with p_{22} THEN (Y is G_{22})

The adaptive network shown in Figure 1 can be used to determine the upper boundary of the probability (1).

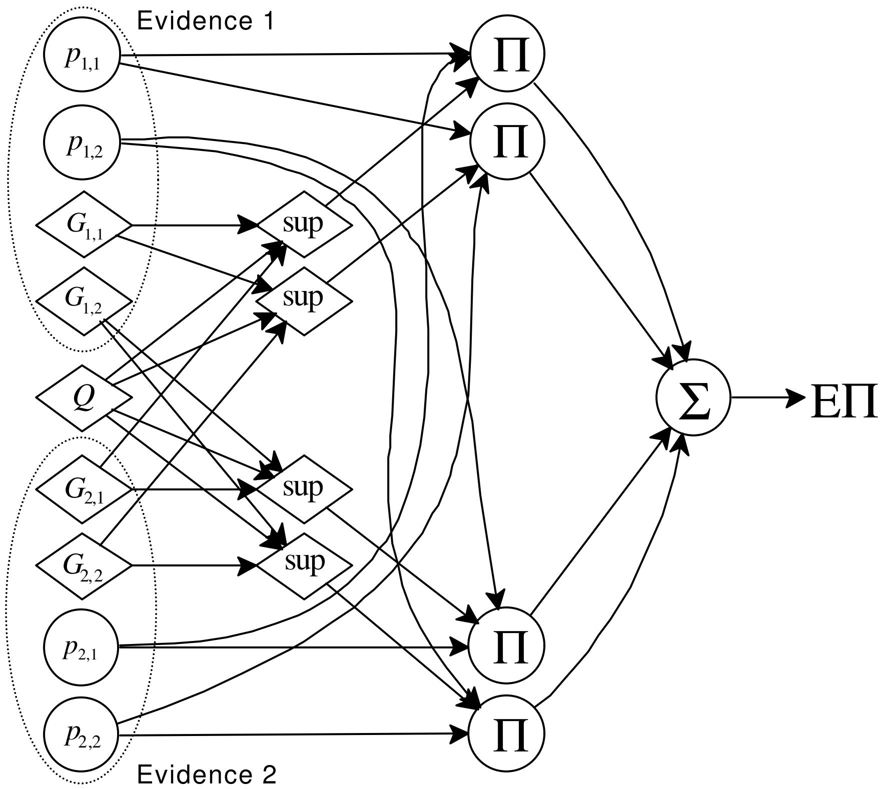

Figure 1. Adaptive network for the calculation of the upper boundary of the probability

Explanation of the meaning of the network processing elements follows.

Oval elements are meant for crisp information processing and diamond elements are meant for fuzzy information processing.

Layer 1:

The output of element $\boldsymbol{G_{ij}}$ is a fuzzy value which corresponds to the right-hand side of the j-th granule in the i-th evidence.

The output of element $\boldsymbol{p_{ij}}$ is the value of the probability associated with the left-hand side of the j-th granule in the i-th evidence.

The output of element $\boldsymbol{Q}$ is the desired value of the criterion.

Layer 2:

The element denoted ***sup*** is used for determining the supremum of the intersection of membership functions fed to the element. Fuzzy values received through dendrites are aggregated by the function of intersection. The transfer function of this element is the supremum function. Thus, the output of this element is the supremum of the intersection of the membership functions received through dendrites.

Layer 3:
The output of element Π is the product of its input values.

Layer 4:
The output of the element Σ is the sum of its input values.

The value of expected certainty *EC* can be determined in a similar manner. The only difference is that instead of element $\boldsymbol{Q}$ there should be element $\boldsymbol{Q'}$, which denotes a complement of fuzzy set Q, and the output of the network should be defined as $(1-\Sigma)$, instead of Σ.

The aforementioned adaptive network is intended for processing the crisp values of probabilities (in a general case, probabilities can be fuzzy or linguistic).

To summarize the aforesaid we can mention the following facts about the ANGIE topology.

The first layer consists of elements that describe a collection of fuzzy bodies of evidence specifying one of the criteria and the desired value of the criterion.

In *the second layer*, the values of the suprema of every possible intersection of the desired value of the criterion and the values of the criteria in the right-hand part of granules are determined.

In *the third layer,* the values of the suprema are multiplied by the corresponding probabilities.

In *the fourth layer*, the values obtained are summed and, as a result, we get the value of $\mathrm{E}\Pi(Q)$.

If there are N bodies of evidence in our collection of bodies of evidence and each one consists of Q_i, $i=1,\ldots,N$ granules, then there are $Q_1\, Q_2\, \ldots\, Q_N$ elements in the second layer. The number of elements in the third layer is equal to $Q_1\, Q_2\, \ldots Q_N$ as well.

4 Training Algorithm

The gradient descent method is used to tune the parameters of ANGIE. That means that in order to tune the parameters we have to determine the partial derivatives of the error function with respect to the parameters we want to tune.

We will use the same error function as in the backpropagation neural networks:

$$\mathrm{E} = \tfrac{1}{2}(d-o)^2,$$

where d is the desired output value and o is the actual value of the system output.

The output of the system is (1). It is obvious that we cannot tune the parameters of the membership functions since the signal is passed through the ***sup*** element.

Calculation of the partial derivatives follows.

According to the chain rule,

$$\frac{dE}{dp_i} = \frac{dE}{do}\frac{do}{dp_i},$$

$$\frac{dE}{do} = (o-d), \quad \text{and} \quad \frac{do}{dp_i} = \sum_j p_{ij} \sup\left(Q \cap G_{1,i} \cap G_{2,j}\right),$$

thus, $$\frac{dE}{dp_i} = (o - d)\sum_j p_{ij} \sup\left(Q \cap G_{1,i} \cap G_{2,j}\right). \tag{3}$$

Again, (3) is for a special case in which there are two bodies of evidence in the system, but it would not present any difficulty to induce a more general type of the formula. It is evident that the value of the partial derivative with respect to the given probability is a sum of terms of $E\Pi(Q)$ which contain the given probability.

In order to keep the probabilities consistent and to make sure their sum is always equal to one, the normalization procedure should be applied to network parameters.

5 Normalizing the Network Parameters

As has been stated earlier, the training algorithm modifies the values of parameters. In its turn, these parameters represent the probabilities of different events. It is obvious that we should make sure that the basic properties of the probabilities hold (e.g. their value should be in the interval *[0, 1]* and they should sum up to one). The algorithm proposed is based on the gradient descent method. This means that the algorithm itself does not even try to make these probabilities consistent with each other. Therefore, the normalization procedure should be applied to network parameters. By normalization we mean a procedure that scales values of the parameters to the interval *[0, 1]* and makes sure that these values sum up to one. There are several possibilities to carry out normalization of system parameters.

It is possible to normalize values of system parameters after each training iteration. This means that we apply the norming procedure after each change in the parameter vales induced by the training algorithm. The advantage of this approach is that we obtain numerically more precise solution. The main drawback is that the normalization procedure requires additional computational power. In addition, normalization will slow down the convergence of the training process. In the worst situation, the training process may diverge if the normalization is applied too often, as the desired output value may not be reachable under the constraint that the sum of the probability values should be equal to one (as we will see, the meaning of probability values is altered during the training).

In order to cope with this problem we may apply the normalization after each *n*-th iteration. In such a way, we can limit our intervention to the training process.

On the other hand, one can apply the normalization procedure only after the training process has finished. The main advantage is evident: we do not interfere the training process, which mean that the convergence time does not slow down. This also implies that the solution may not be precise numerically.

How important is the numerical accuracy of the obtained probability values? After the training of the system, the probability values indicate the *importance* of the corresponding fuzzy value (feature) for the criterion to take the desired value. This means that features with higher probability values (obtained after the training) are more preferable under the given desired value of the criterion. Thus, the probability values can be considered as *relative importance coefficients* of the corresponding fuzzy features. This means that their relative and not the absolute values do matter. Hence, it can be concluded that the latter approach to normalization is more preferable, as it enables one to save the computational resources.

5.1 Case Study

Let us consider an example of using the adaptive network ANGIE for sensitivity analysis. The case study we are going to examine is more closely related to the everyday decisions we are to make, rather than to the "usual" problems considered in the context of decision analysis.

Nevertheless, from the computational point of view, this problem is far from trivial. We assume that we have to determine the influence of different gift parameters on the evaluation of a gift. This problem is considered in detail in [4]. The problem is similar to sensitivity analysis. However, in the traditional sensitivity analysis only one or two parameters are taken into consideration as the result is represented graphically. In this case study we will try to determine the relationship among about 20 different values of parameters. However, it should be remembered that the sensitivity analysis is just one step in the decision aid model analysis and rarely leads directly to the final decision.

The gift is evaluated using the criterion *"Y = Evaluation of the gift"* that assumes fuzzy values on some gift evaluation scale. Gifts are described using six parameters: price, colour, weight, expenses that appear during the exploitation and the parameter that defines for how long the gift can be used without expenses.

Before starting the training process, we have to define bodies of granular evidence that describe possible values of the above-mentioned parameters. The system consists of six bodies of evidence, as there are six parameters. The possibility of using adaptive network ANGIE for sensitivity analysis is described in more detail in [4, 5].

The problem considered is quite bulky from the computational point of view. There are *7250* processing elements and *25200* links in the system. This may explain why the training process has diverged in the case when the normalization procedure has been applied after each iteration. Learning speed during the training process is equal to *0.3*.

The parameters of the first system have been normalized at every fifth iteration. The convergence line (the bold line) and values of system parameters during the training are shown in Figure 2. The system has converged on *140th* iteration.

The parameters of the second system have been normalized only after the training process has converged. The convergence line (the bold line) and values of system parameters during the training are shown in Figure 3. The system has converged at *78th* iteration.

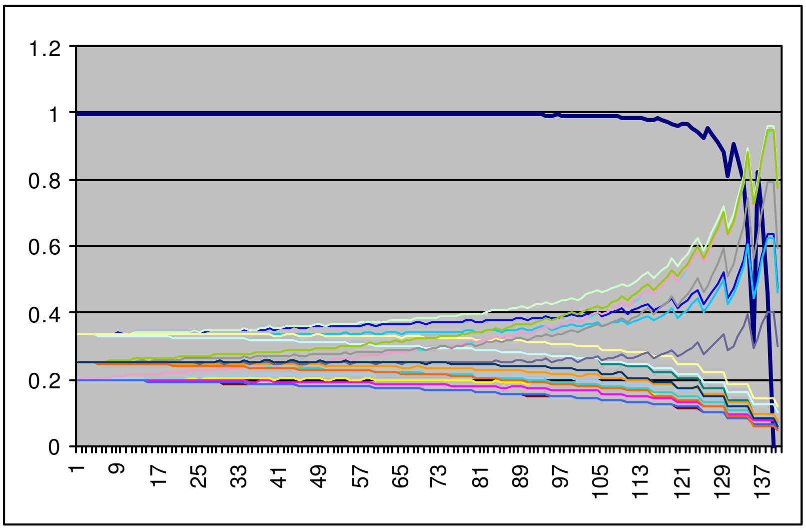

Figure 2. Convergence line and network parameters, normalization at every 5th iteration

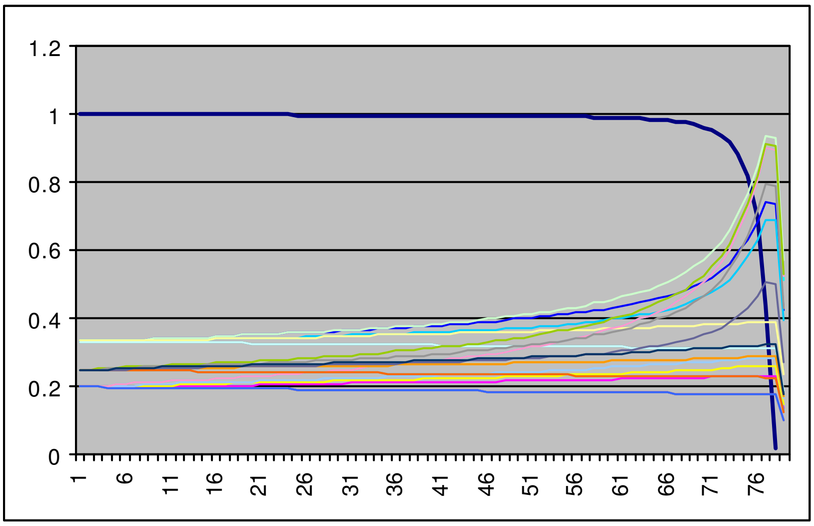

Figure 3. Convergence line and network parameters, normalization after termination of the training process

As can be seen, if the ANGIE network parameters are not normalized during the training process, the convergence is about two times faster. Moreover, we save some time by not running the normalization procedure, but given that the normalization complexity is linear, it does not have a big impact on the overall performance, as the size of the network is combinatorial (refer to the formula (1)). In addition, in [4] it has been shown that the solutions obtained by the first and by the second system are equivalent.

6 Conclusion

In this paper granular evidence-based adaptive network ANGIE has been considered. The network can be used in fuzzy granule-based decision aid models. Training algorithm can be derived for this network, which enables one to use the considered network to do multi-parameter sensitivity analysis tasks. One of the training algorithm issues concerns normalization of the network parameters, as they represent probability values and their consistency should be preserved.

There are three possible ways to carry out the normalization procedure. First, one can apply normalization after every iteration, second, one can apply normalization each n-th iteration or third, one can normalize the network parameter values after the training process has converged (thus, not influencing the training process by normalization of the parameters).

Empirical results show that normalizing the network parameters after each iteration may lead to slow convergence or to divergence of the training process. Normalizing the network parameters after each n-th iteration allows one to increase the convergence speed. However, it has been stated that the relative and not the absolute accuracy of the network parameter values does matter. This implies that normalization of the network parameters after the training algorithm convergence is more preferable.

The future direction of the research commenced in this paper, which is already in progress, is the study of the possibility of applying granular computations in the field of risk analysis. The idea of this research is to study the possibility of evaluating riskiness of different alternatives described with the help of granular bodies of evidence.

References

[1] J.-S. R. Jang, ANFIS: Adaptive-Network-Based Fuzzy Inference System, *IEEE Transactions on Systems, Man and Cybernetics*, Vol. 23, No. 3, 665-684, May/June 1993.

[2] R.T. Clement, T. Reilly, *Making Hard Decisions*, Duxbury, USA 1996.

[3] L.A. Zadeh, Fuzzy Sets and Information Granularity, *Advances in Fuzzy Set Theory and Applications*, Gupta M.M., Ragade R.K et al. (Eds), North-Holland Publishing Company, 3-18, 1979.

[4] A. Valishevsky, A. Borisov, Using Granular-Evidence-Based Adaptive Networks for Sensitivity Analysis, *Scientific Proceedings of Riga Technical University*, Vol. 10, RTU, Riga, Latvia, 150-156, 2002.

[5] A. Valishevsky, A. Borisov, ANGIE: Adaptive Network for Granular Information and Evidence Processing, *Fifth International Conference on Application of Fuzzy Systems and Soft Computing ICAFS-2002,* Milan, Italy, 166-173, September 2002.