

# A Fuzzy ARTMAP Probability Estimator with Relevance Factor

Răzvan Andonie, Lucian Sasu  
Transylvania University, 2200 Braşov, Romania

**Abstract.** An incremental, nonparametric probability estimation procedure using a variation of the Fuzzy ARTMAP (FAM) neural network is introduced. The resulted network, called Fuzzy ARTMAP with Relevance factor (FAMR), uses a relevance factor assigned to each sample pair, proportional to the importance of that pair during the learning phase. We prove that our probability estimator is correct. The FAMR can be used both as a classifier and as a probability estimator.

## 1 Introduction

In the context of supervised training, *incremental learning* means learning each input-output sample pair, without keeping it for subsequent processing.

Many pattern recognition applications require an estimate of the *posterior* probability  $P(C|\mathbf{a})$ , where  $C$  is a class index and  $\mathbf{a}$  is an input pattern. This task also allows classification because one can select the class  $C$  with maximum conditional probability.

This paper will discuss the probability estimation from data samples in supervised incremental learning systems based on Fuzzy ARTMAP (FAM) architectures. Such procedures are presented in [3, 7, 5, 6].

This paper introduces a variation of the probability estimation phase of FAM and identify the resulted network as FAMR to distinguish it from the original architecture. FAMR is an incremental learning system for general classification and nonparametric estimation of the probability that an input belongs to a given class. Each training pair has a *relevance factor* assigned to it. This factor is proportional to the importance of that pair during the learning phase. Using a relevance factor adds more flexibility to the training phase, allowing ranking of sample pairs according to the confidence we have in the information source. The training sequence may include sample pairs from sources with different levels of noise.

In Section 2, we briefly discuss how the FAM architecture was used for probability estimation. Section 3 introduces our modification of the FAM algorithm. In Section 4 we present the experimental results and finally, in Section 5 we conclude with some closing remarks.

## 2 Probability estimation in FAM

A detailed description of the FAM probability estimation can be found in [3]. We present here only the necessary details.

FAM includes a pair of ART modules ( $ART_a$  and  $ART_b$ ) that create stable recognition categories in response to arbitrary sequences of input patterns. These modules are linked by an inter-ART module called Mapfield whose purpose is to determine whether the correct mapping has been established from inputs to outputs or not.

During learning, FAM updates its Mapfield weights to estimate the probability that an input belongs to a given output class: the strength of the weight projecting from the selected  $ART_a$  category to the correct  $ART_b$  category is increased, while the strength of the weights to other  $ART_b$  categories are decreased. A Mapfield vigilance parameter  $\rho_{ab}$  calibrates the degree of predictive mismatch, necessary to trigger the search for a different  $ART_a$  category. If the weight projecting from the active  $ART_a$  category through the Mapfield to the active  $ART_b$  category is smaller than  $\rho_{ab}$  (vigilance test), the system responds to the unexpected outcome through the so-called *match tracking*, which triggers an  $ART_a$  search for a new input category.

Once an  $ART_a$  category  $J$  is chosen, whose prediction of the correct  $ART_b$  category is strong enough, match tracking is disengaged, and the network is said to be in a state of resonance. In this case, Mapfield learns by updating the weights of associations between  $ART_a$  and  $ART_b$  categories. According to this updating scheme, weight  $w_{jk}^{ab}$  is a non-decreasing function of the frequency of associations between the  $j$ th  $ART_a$  category and the  $k$ th  $ART_b$  category during the training phase.

This last feature is made more explicit in PROBART [7], where Mapfield weight  $w_{jk}^{ab}$  is exactly the frequency of associations between the  $j$ th  $ART_a$  category and the  $k$ th  $ART_b$  category. Therefore,  $w_{jk}^{ab}/|\mathbf{w}_j^{ab}|$  is the empirical estimate of the posterior probability  $P(k|j)$  that  $ART_a$  category  $j$  is associated to  $ART_b$  category  $k$ .

## 3 The FAMR Algorithm

Let us consider a sequence of independent experiments according to the finite probability distribution  $P(a_1), \dots, P(a_n)$ , where  $P(a_i) > 0$  is the probability of outcome  $a_i$ ,  $\sum_{i=1}^n P(a_i) = 1$ . These *objective probabilities* are not known and will be estimated at each step based on the previous observations. A criterion for a qualitative differentiation of the experiments is represented by the relevance associated to each experiment. The *relevance*  $q_t$  is a real positive finite number directly proportional to the importance of the experiment considered at step  $t$ . This number may be either of objective or subjective nature. The following estimation procedure (defined in [1]) makes use both of the results and the relevances of the present and previous experiments.

The *subjective probability* of outcome  $a_i$  ( $i = 1, \dots, n$ ) at step  $t$  ( $t = 1, 2, \dots$ )

is given by:

$$w_t(a_i) = w_{t-1}(a_i) + A_t (\delta_t(a_i) - w_{t-1}(a_i)) \quad (1)$$

where: if at step  $t$  we get outcome  $a_j$ ,  $\delta_t(a_j) = 1$  and  $\delta_t(a_i) = 0$  for  $j \neq i$ ;  $w_0(a_i) > 0$  is the initial subjective probability,  $\sum_{i=1}^n w_0(a_i) = 1$ ;  $q_0$  is the initial relevance,  $Q_t = \sum_{s=0}^t q_s$ ,  $A_t = q_t/Q_t$ .

**Theorem 1.**  $w_t(a_i) \xrightarrow{t} P(a_i)$  in probability iff  $Q_t \xrightarrow{t} \infty$ .

For some additional conditions imposed to  $q_t$ , the direct result can be strengthened:

**Theorem 2.** If  $q_0 \in [0, b]$ ,  $q_t \in [a, b]$  ( $t = 1, 2, \dots$ ), for two real values  $0 < a \leq b < \infty$ , then  $w_t(a_i) \xrightarrow{t} P(a_i)$  with probability one.

In other words, an observer who intends to learn objective probabilities from examples has to have sufficient confidence in the results of the experiences. Theorem 1 is from [1], whereas Theorem 2 is new.

Mapfield weight  $w_{jk}^{ab}$  can be considered an estimate of the posterior probability  $P(k|j)$ . This enables us to use formula (1) to update the weights  $w_{jk}^{ab}$ :

$$w_{Jk}^{ab(new)} = \begin{cases} w_{jk}^{ab(old)} & \text{if } j \neq J \\ w_{JK}^{ab(old)} + A_t(1 - w_{JK}^{ab(old)}) & \\ w_{Jk}^{ab(old)}(1 - A_t) & \text{if } k \neq K \end{cases} \quad (2)$$

Is  $w_{jk}^{ab}$  a good estimate of  $P(I_b|I_a)$ , where  $I_a$  and  $I_b$  are intervals based around input pattern  $\mathbf{a}$ , respectively output pattern  $\mathbf{b}$ ? Feedback via match tracking alters this estimation (see [7]). One way to avoid this problem is to eliminate match tracking.

If the conditions in Theorem 2 are true and match tracking is not used then, for each  $ART_a$  category  $j$  ( $j = 1, \dots, N_a$ ) and each  $ART_b$  category  $k$  ( $k = 1, \dots, N_b$ ), we have:

$$w_{jk}^{ab} \rightarrow P(k|j) \text{ with probability one.} \quad (3)$$

Eliminating match tracking is not always convenient, because match tracking controls category proliferation in  $ART_a$ . Meanwhile, it is difficult to say something about this probability approximation in the presence of match tracking, since in this case  $w_{jk}^{ab}$  is not necessarily a good estimate of the posterior probability with respect to the already processed data. However, in our experiments, match tracking has not significantly altered probability estimation.

Let  $\mathbf{Q}$  be the vector  $[Q_1 \dots Q_{N_a}]$ .  $N_a$  and  $N_b$  are the number of categories in  $ART_a$ , respectively  $ART_b$ , initialized with 0. For incremental learning of one training pair, the new Mapfield algorithm is given in Algorithm 1.

Since we initialize the weights  $w_{jk}^{ab}$  with  $1/N_b$  and not with 1, we have to modify the vigilance test. The new test is:

$$N_b w_{JK}^{ab} \geq \rho_{ab} \quad (4)$$

*Step 1.* Accept vector pair  $(\mathbf{a}, \mathbf{b})$  with relevance factor  $q$ .  
*Step 2.* If necessary, create category  $K$  in  $ART_b$ :  
 $N_b = N_b + 1$   
 $K = N_b$   
**if**  $N_b > 1$  **then**  
 $w_{jK}^{ab} = \frac{q_0}{N_b Q_j}$  for  $j = 1, \dots, N_a$  {append new component to  $\mathbf{w}_j^{ab}$ }  
 $w_{jk}^{ab} = w_{jk}^{ab} - \frac{w_{jK}^{ab}}{N_b - 1}$  for  $k = 1, \dots, K - 1; j = 1, \dots, N_a$  {normalize}  
**endif**  
*Step 3.* If necessary, create category  $J$  in  $ART_a$ :  
 $N_a = N_a + 1$   
 $J = N_a$   
 $Q_J = q_0$  {append new component to  $\mathbf{Q}$ }  
 $w_{jK}^{ab} = 1/N_b$  for  $k = 1, \dots, N_b$  {append new line to  $\mathbf{w}^{ab}$ }  
*Step 4.*  $J, K$  are winners or new added nodes  
**if** vigilance test (4) is passed **then**  
{learn in Mapfield}  
 $Q_J = Q_J + q$   
 $w_{JK}^{ab} = w_{JK}^{ab} + \frac{q}{Q_J}(1 - w_{JK}^{ab})$   
 $w_{Jk}^{ab} = w_{Jk}^{ab} \left(1 - \frac{q}{Q_J}\right)$  for  $k = 1, \dots, N_b, k \neq K$   
**else**  
perform match tracking and restart from step 3  
**endif**

Algorithm 1: **One iteration in the new Mapfield algorithm.**

The rest of the FAM mechanism remains unchanged. The resulted algorithm will be called FAMR (Fuzzy ARTMAP with Relevance factor).

Using a relevance factor in FAMR is not equivalent to repeatedly presenting a training sample to the system: the variation of  $w_{JK}^{ab}$  values is finer than in the case of repeating the presentation of the training pair, since the relevance factor can be a real value. Second, learning is faster, because we can learn in one step instead of repeatedly learning the same pair.

How to assign a relevance factor to a training sample? An answer could be in ranking the sample pairs according to the confidence we have in the information source. We have in mind at least two application areas for such learning systems with relevance factor.

1. When training neural networks with noisy data, we can assign a relevance factor inverse proportional to the noise.

2. Assuming we can generate training pairs close to the decision boundary, we could assign a relative higher relevance factor to this samples. However, there are experimental results reported [4] showing that choosing examples from the boundary area does not necessarily conduct to better classification performances. That remains an open area for further investigations.

## 4 Experimental Results

A suite of experiments were performed to test the FAMR's aptitude for probability estimation and classification. Only incremental learning was used, even if the network is able to improve its performance using off-line processing, when the training set is reprocessed. The classification was made based on the probability estimation by hard-decision: an input pattern belongs to the category with maximum posterior probability. The performance of probability estimator was quantified by the average Brier score. This score measures the quality of probability estimation by comparing it to the objective conditional probability [3].

**I. Circle-in-the-square.** This problem requires a system to identify which points of a square lie inside and which lie outside a circle whose area equals half that of the square. We have two classes of points: points located inside the circle and points located outside the circle. For computing the Brier score, 1000000 evenly spaced points were generated inside the square. The relevance factor was set here to 1.

The training set contained 1000, 10000, and 100000 patterns. The test set consisted of 100000 patterns in each case. As expected, the recognition rate and the Brier score increased with the number of training patterns from an average value of 92.995% and 0.9327 respectively (for 1000 training patterns) to 98.101% and 0.9810 (for 100000 training patterns). Compared to the results reported in [2] we have obtained on an average better performances: less or equal number of  $ART_a$  categories, better Brier score and recognition rate.

**II. Noisy circle-in-the-square.** A modified version of the circle-in-the-square problem was used in order to test the effectiveness of the relevance factor. We considered three data sources (called  $A$ ,  $B$ ,  $C$ ), each of them producing the same number of training samples. Each source has an associated probability ( $p_A$ ,  $p_B$ , and  $p_C$  respectively) of producing wrong associations. We took  $(p_A, p_B, p_C) = (0, 0.2, 0.35)$ . First, the relevance factor  $q_t$  was set to 1, for each information source. The average Brier score obtained for 6 different sets of data was 0.89567. Subsequently, we considered different relevance factors, in accordance to the noise level of the three sources:  $(q_A, q_B, q_C) = (100, 10, 1)$ , where  $q_X$  is the relevance factor associated with the data source  $X$ . The average Brier score obtained for 6 different sets of data was 0.91895, superior to the the previous case. The total number of training patterns was 10000 for each experiment and the Brier score was computed for 10000 points evenly distributed inside the square.

Correlating the relevance factors to the degree of confidence in each data source resulted in a better performance of the system. The relatively small value of the average Brier score is explained by the presence of noise.

In order to prove the advantage of taking into account supplementary data sources, though these sources were noisy, we developed another experiment. This experiment proved more relevant when the number of available correct training samples was relatively small. First, we have generated 1000 associations using three data sources ( $A$ ,  $B$ ,  $C$ ), each with the same

probability of producing training patterns,  $(p_A, p_B, p_C) = (0, 0.2, 0.35)$ , and  $(q_A, q_B, q_C) = (100, 10, 1)$ . The average Brier score for different training sets was 0.88370 for 1000 training patterns, above 0.88033, the value obtained when using only the 1000/3 correct samples from source A to train the FAMR.

## 5 Conclusions

The Mapfield algorithm developed here expands the range of FAM applications by allowing to assign a relevance factor to each training pair. The FAMR probability estimation is simple and converges with probability one to the posterior probability. Compared to the FAM probability estimator, FAMR shows similar or better performances with respect to the Brier score, recognition rate, and number of generated nodes. The true benefits of using FAMR may come from using a relevance factor assigned to the training samples.

## References

- [1] Răzvan Andonie. A converse H-theorem for inductive processes. *Computers and Artificial Intelligence*, Vol. 9:159–167, 1990.
- [2] G.A. Carpenter, S. Grossberg, N. Markuzon, J.H. Reynolds, and D.B. Rosen. Fuzzy ARTMAP: A neural network architecture for incremental supervised learning of analog multidimensional maps. *IEEE Transactions on Neural Networks*, 3(5):698–713, 1992.
- [3] G.A. Carpenter, S. Grossberg, and J.H. Reynolds. A fuzzy ARTMAP nonparametric probability estimator for nonstationary pattern recognition problems. *IEEE Transactions on Neural Networks*, 6(6):1330–1336, 1995.
- [4] V. Ciesielski. Boundary points do not improve the accuracy of neural net classifiers. In *Proceedings of the Eighth Australian Joint Conference on Artificial Intelligence*, pages 163–170, Canberra, 1995.
- [5] C. P. Lim and R. F. Harrison. An incremental adaptive network for on-line supervised learning and probability estimation. *Neural Networks*, 10(5):925–939, 1997.
- [6] C. P. Lim and R. F. Harrison. ART-Based Autonomous Learning Systems: Part I - Architectures and Algorithms. In L. C. Jain, B. Lazzerini, and U. Halici, editors, *Innovations in ART Neural Networks*. Springer, 2000.
- [7] S. Marriott and R. F. Harrison. A modified fuzzy ARTMAP architecture for the approximation of noisy mappings. *Neural Networks*, 8(4):619–641, 1995.