Fuzzy ARTMAP with Relevance Factor

Răzvan Andonie
Department of Electronics and Computers
Transylvania University of Braşov
Email: andonie@deltanet.ro

Lucian Sasu
Department of Computer Science
Transylvania University of Braşov
Email: lmsasu@unitbv.ro

Valeriu Beiu
School of EE and CS
Washington State University, Pullman
Email: vbeiu@eecs.wsu.edu

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 the respective pair during the learning phase. Experimental results have shown that FAMR favorably compares with FAM and Probabilistic FAM (PFAM, defined in [1], [2]), both as a classifier and as a probability estimator.**

I. Introduction

When designing and implementing data mining applications for large data sets, we face processing time and memory space problems. In this case, incremental learning is a very attractive feature. According to [3], we define an *incremental learning* algorithm as one that meets the following criteria:

1) It should be able to learn additional information from new data.
2) It should not require access to the original data, used to train the existing system.
3) It should preserve previously acquired knowledge.
4) It should be able to accommodate new data categories that may be introduced with new data.

The fundamental issue in incremental learning is: how can a learning system adapt to new information without corrupting or forgetting previously learned information – the so-called *stability-plasticity* dilemma addressed by Carpenter and Grossberg [4].

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

The topic addressed in this paper is the development of a supervised incremental learning algorithm satisfying all of the above-mentioned criteria. Very few algorithms perfectly fit into this description of incremental learning. The FAM family of neural networks, having the roots in Carpenter, Grossberg, Markuzon, Reynolds, and Rosen's seminal paper [5] is the best known example. A more recent neural network having this strong property is described by Polikar, Udpa, Udpa, and Honovar [3].

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 the maximum conditional probability.

The present paper only deals with the posterior probability estimation from data samples in supervised incremental learning systems based on FAM architectures. Such procedures have been developed by Carpenter, Grossberg, and Reynolds [6], and Marriott and Harrison [7].

Lim and Harrison's PFAM [1], [2] is a hybrid FAM + Probabilistic Neural Network (PNN, see [8]) classifier with incremental probability estimation capabilities: It uses the PNN's ability to incrementally construct an approximation of the probability density functions (pdf) and it also uses the code compression feature of FAM. Instead of considering every sample pattern in estimating pdf, the clustering property of FAM is used to obtain the centroid of each cluster. The pdf approximation is made based on these centroids only.

This paper introduces a variation of the probability estimation phase of FAM and identifies 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. The architecture of the network is able to incrementally 'grow' and to sequentially accommodate input-output sample pairs. Each training pair has a *relevance factor* assigned to it. This factor is proportional to the importance of the respective pair in the learning process. 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.

Experimental results have demonstrated that FAMR favorably compares with FAM and PFAM, both as a classifier and as a probability estimator.

In Section II, we briefly discuss how the FAM architecture was used for probability estimation. Section III introduces our modification of the FAM algorithm. In Section IV we present the experimental results comparing the FAMR model to FAM and PFAM. Section V concludes with some closing remarks.

II. FAM as an Incremental Probability Estimator

Carpenter, Grossberg, and Reynolds' FAM [6] can estimate posterior probabilities via formation and associations between intermediate categories. 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

0-7803-7898-9/03/$17.00 ©2003 IEEE

from inputs to outputs or not. The ART_a and ART_b vigilance parameters ρ_a, respectively ρ_b, control the matching mechanism inside the modules.

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 ρ_{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 ρ_{ab} (vigilance test), then the system responds to the unexpected outcome through the so-called *match tracking*, that 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 resonance state. 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 jth ART_a category and the kth 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 jth ART_a category and the kth 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.

III. The FAMR Algorithm

A. A probability estimation procedure

A stochastic approximation procedure described in [9] is introduced and new theoretical results are developed. Let us consider a sequence of independent experiments according to the finite probability distribution $P(a_1), \ldots, P(a_n)$, where $P(a_i) \geq 0$ 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 $(t = 1, 2, \ldots)$. This number may be either of objective or subjective nature.

The following estimation procedure makes use of both the results and the relevances of the present, and previous experiments.

The *subjective probability* of outcome a_i $(i = 1, \ldots, n)$ at step t $(t = 1, 2, \ldots)$ is given by:

$$w_t(a_i) = \frac{\left(q_0 w_0(a_i) + \sum_{s=1}^{t} q_s \delta_s(a_i)\right)}{Q_t} \qquad (1)$$

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

At each step t $(t = 0, 1, \ldots)$ we have a probability vector with $w_t(a_i) \geq 0$ $(i = 1, \ldots, n)$, $\sum_{i=1}^{n} w_t(a_i) = 1$.

Relation (1) can be rewritten in a recursive form:

$$w_t(a_i) = w_{t-1}(a_i) + A_t\left(\delta_t(a_i) - w_{t-1}(a_i)\right) \qquad (2)$$

where $A_t = q_t/Q_t$ $(t = 1, 2, \ldots)$. The following result is from [9]:

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

Consequently, $w_t(a_i)$ is a correct biased estimator of $P(a_i)$ iff $Q_t \to \infty$. Further analysis of the estimate can be made if we compute the mean square error:

$$\alpha_t(a_i) = (1 - A_t)^2 \alpha_{t-1}(a_i) + P(a_i)(1 - P(a_i))\, A_t^2 \qquad (3)$$

where $\alpha_t(a_i) = E(w_t(a_i) - P(a_i))^2$. This expression gives us the possibility of evaluating the rate of convergence.

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

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

Sketch of proof: Equation (2) can be rewritten as a Robbins–Monroe process. The proof is based on the Stochastic Approximation Theorem.

In practice, the above restriction imposed to q_t does not restrict our estimation procedure. The meaning of the conditions in the previous theorems is: an observer who intends to learn objective probabilities from examples has to have sufficient confidence in the results of the experiences.

Let $w_t^{(n)}(a_i)$ be the subjective probabilities at step t $(t = 1, 2, \ldots)$, for n possible outcomes. What is happening if at some step we get a new outcome, a_{n+1}? Assuming we have $w_0^{(n)}(a_i) = 1/n$ $(i = 1, \ldots, n)$, then the new subjective probabilities $w_t^{(n+1)}(a_i)$ for $n+1$ possible outcomes may be obtained by the following relations:

$$\begin{aligned} &w_t^{(n+1)}(a_{n+1}) = q_0/(n+1)Q_t \\ &w_t^{(n+1)}(a_i) = w_t^{(n)}(a_i) - w_t^{(n+1)}(a_{n+1})/n,\ i = 1, 2, \ldots, n \end{aligned} \qquad (4)$$

Relations (4) will be used in the dynamic allocation of ART_b categories (*Step 2* in Algorithm 1.)

B. The FAM modification

A modification of the FAM, named FAMR, that enhances the probability estimation ability of FAM is presented.

Mapfield weight w_{jk}^{ab} can be considered an estimate of the posterior probability $P(k|j)$. This enables us to use formula (2) to update the weights w_{jk}^{ab}:

$$w_{Jk}^{ab(new)} = \begin{cases} w_{jk}^{ab(old)} & \text{if } j \neq J \\ w_{JK}^{ab(old)} + A_t(1 - w_{JK}^{ab(old)}) & \\ w_{Jk}^{ab(old)}(1 - A_t) & \text{if } k \neq K \end{cases} \qquad (5)$$

Is w_{jk}^{ab} a good estimate of $P(I_\mathbf{b}|I_\mathbf{a})$, where $I_\mathbf{a}$ and $I_\mathbf{b}$ are intervals based around input pattern a and output pattern b, respectively? As depicted by Marriott and Harrison the feedback via match tracking alters this estimation [7]. One way to avoid this problem is to eliminate match tracking. This approach is used in PROBART and ensures that a given input to ART_a will always select the same category. Meanwhile, eliminating match tracking allows for one-to-many mapping between ART_a and ART_b categories, which may be important in situations where more than one action result from a single input [7].

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

$$w_{jk}^{ab} \rightarrow P(k|j) \text{ with probability one.} \quad (6)$$

Match tracking can be avoided by setting $\rho_{ab} = 0$. Eliminating match tracking is not always convenient, because match tracking controls category proliferation in ART_a. On the other hand, one could hardly say anything 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. A smaller value for ρ_{ab} results in a better approximation. For $\rho_{ab} = 0$ the approximation is statistically correct. However, in our experiments, match tracking has not significantly altered probability estimation.

Let $\mathbf{Q}$ be the vector $[Q_1 \ldots Q_{N_a}]$. N_a and N_b are the number of categories in ART_a and ART_b, initialized to 0, respectively. For incremental learning of one training pair, the new procedure in Mapfield 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:

$$N_b \, w_{JK}^{ab} \geq \rho_{ab} \quad (7)$$

The rest of the FAM mechanism remains unchanged. The resulted algorithm will be called FAMR (Fuzzy Artmap with Relevance factor.) In [10], we have introduced a probability estimator based on a restricted FAMR version, where estimated probabilities are strictly positive.

For $\rho_{ab} = 0$ (no match tracking), $q_0 = 0$, $q_t = q$, $0 < q < \infty$ ($t = 1, 2, \ldots$), probability estimate w_{jk}^{ab} is exactly the empirical estimate of the posterior probability $P(k|j)$. This can be observed from the nonrecursive formula (1). Therefore, PROBART is a particular case of FAMR.

In our experiments, since we have used relatively large training sets, the influence of the initial values (probabilities and relevance) was insignificant. We have set $q_0 = 1$ for all experiments. The initial probabilities in Algorithm 1 are equal. Generally, the initial values can influence the stability of the system (i.e., how fast it learns), especially for the first iterations.

Step 1. Accept vector pair (a, 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 $j = 1, \ldots, N_a$
    {append new component to $\mathbf{w}_j^{ab}$}
    $w_{jk}^{ab} = w_{jk}^{ab} - \frac{w_{jK}^{ab}}{N_b - 1}$ for $k = 1, \ldots, K-1$,
    $j = 1, \ldots N_a$ {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 $k = 1, \ldots, N_b$
  {append new line to $\mathbf{w}^{ab}$}
Step 4. J, K are winners or new added nodes.
  if vigilance test (7) 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 $k = 1, \ldots, N_b$, $k \neq K$
  else
    perform match tracking and restart from step 3
  endif

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

C. Application areas of the relevance factor

Ranking the importance of training examples in neural computing has been considered by several authors. Gallant uses an importance factor attached to each training sample [11]. Proportional to the importance factor, additional duplicates of each training sample are created.

In FAMR, using a relevance factor is not equivalent to repeatedly present 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 reside in ranking the sample pairs according to the (subjective) confidence we have in the information source. Two application areas are considered for such learning systems with relevance factor:

1. When training neural networks with noisy data, a relevance factor could be assigned to each learning pattern, inversely proportional to the noise. Let us suppose that we have a training sequence consisting of two sample pairs: $(\mathbf{a}_1 = 0.1, class\_index(\mathbf{a}_1) = 1)$ with $q_1 = 1$, and $(\mathbf{a}_2 = 0.3, class\_index(\mathbf{a}_2) = 2)$ with $q_2 = 1$. We assume that $class\_index(\mathbf{a}_1)$ is a correct association, whereas $class\_index(\mathbf{a}_2)$ is a noisy association (that should be 1.) After two iterations in the FAMR algorithm, assuming that

only one ART_a category is generated, the new probability vector will be

$$\mathbf{w}_1^{ab} = [0.5 \;\; 0.5] \tag{8}$$

If we perform FAMR training with $q_1 = 2$ and $q_2 = 1$ (the first pair is more relevant than the second one), we obtain:

$$\mathbf{w}_1^{ab} = [0.62 \;\; 0.37] \tag{9}$$

Let us classify pattern $\mathbf{a}_2$. The second trained network makes a better prediction, indicating class 1 with the highest probability. In this example, the relevance factor acts as a noise filter.

2. Training pairs are usually randomly selected. However, it seems reasonable to expect that if correctly classified examples are chosen near the decision boundaries then the classifier will learn the boundaries better. This conjecture has not been significantly explored, most probably because the true boundaries are usually unknown at the beginning of the training. Assuming we can generate points close to the boundary, we could assign a relative higher relevance factor to this samples. There are experimental results reported [12] showing that choosing examples from the boundary area does not necessarily lead to better classification performances. That remains an open area for further investigation.

IV. Experiments and Results

A suite of experiments were performed to test the FAMR's ability for probability estimation and classification, compared to FAM and PFAM. 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 the probability estimator was quantified by an average Brier score. The Brier score measures the quality of the probability estimation by comparing it to the real conditional probability [6]. The score $u(q,p)$ is a function of the estimated probability q and the true probability p:

$$u(q,p) = 1 - (q-p)^2 \tag{10}$$

We have used only incremental learning, though the network is able to improve its performance using off-line processing, when the training set is reprocessed, or using Multiple Classifier Systems. Unless otherwise specified, the used relevance factor was 1. In the prediction phase, we took $\rho_a = 0$; thus, any input pattern is assigned to an ART_a category and subsequently to an output class.

A. 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. Patterns were generated inside the square using an uniform distribution for each coordinate. The points were classified according to their position relative to the circle, whose center coincides with the center of the square. Thus 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.

TABLE I

Circle-in-the-square: Average values of ART_a categories number and test set recognition rate for FAMR compared to results from [5]. The FAMR results represent average values for 5 different training sets.

Train size	ART_a categories number		Test set recognition rate (%)	
	FAMR	Carpenter [5]	FAMR	Carpenter [5]
1000	18.2	21	93.0	92.5
10000	45.2	50	96.8	96.7
100000	111.6	121	98.1	98.0

The training sets contained 1000, 10000, and 100000 patterns. The test set consisted of 100000 patterns in each case. For each training set size, five different training sets were generated and the average Brier score was computed at the end of every training phase. The number of ART_a categories was at most as large as reported in [5], but the performance was superior. The results for the three training sets are presented in Table I. As expected, the test set recognition rate and the Brier score increased with the number of training patterns from an average value of 93.0% and 0.9327 (for 1000 training patterns) to 98.1% and 0.9810, respectively (for 100000 training patterns.)

B. Noisy circle-in-the-square

We used a modified version of the circle-in-the-square problem 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 data sets was 0.89568. 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 the 6 different data sets was 0.91896, higher than the previous case (Table II.) 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.

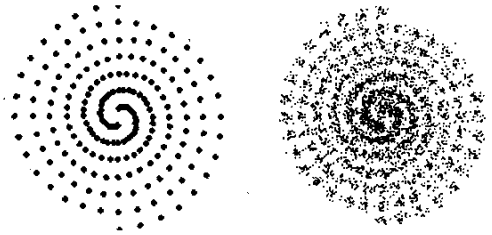
Correlating the relevance factors to the degree of confidence in each data source resulted in higher performances for the system. The relatively small value of the average Brier score is explained by the presence of noise.

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

TABLE II

AVERAGE BRIER SCORE FOR NOISY CIRCLE–IN–THE–SQUARE ASSOCIATIONS. $(p_A, p_B, p_C) = (0, 0.2, 0.35)$, WHERE p_X IS THE PROBABILITY THAT DATA SOURCE X GIVES WRONG ASSOCIATIONS. q_X IS THE RELEVANCE FACTOR ASSOCIATED WITH DATA SOURCE X.

Test no.	Average Brier score	
	$(q_A, q_B, q_C) = (100, 10, 1)$	$(q_A, q_B, q_C) = (1, 1, 1)$
1	0.92164	0.89810
2	0.91672	0.89251
3	0.93540	0.90876
4	0.91018	0.88908
5	0.91298	0.89215
6	0.91682	0.89346
Average	0.91896	0.89568



(a) Nested spirals: training set.

(b) Nested spirals: test set

Fig. 1. Two nested spirals

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.

C. Learning to tell two spirals apart

The two spirals [13] make three complete turns in the plane, totaling 194 points (the training set.) For the test set, we added Gaussian noise centered in each point, with standard deviation 0.1. The train and the test set are represented in Fig. 1(a) and Fig. 1(b), respectively.

Each Gaussian cluster contains 20 points giving a total number of 3880 test patterns. The number of ART_a categories is 82, and the test set recognition rate has an average value of 94.55% (using five differents test sets), while the clusters are fairly close. As justified in [6], the Brier score is an underestimate of FAMR performance because it does not reflect the network's ability of recomposing the complex underlying geometrical shape.

D. Two Gaussians

This test [6] consists of estimating the posterior probability of input patterns from two normally distributed overlapping classes (Fig. 2.) The input points are located inside the unit square and they are drawn from two Gaussian distributions centered in $\mu_1 = (0.5, 0.75)$ and $\mu_2 = (0.5, 0.25)$, with covariance matrix

$$\Sigma = \begin{pmatrix} 0.15^2 & 0 \\ 0 & 0.15^2 \end{pmatrix} \tag{11}$$

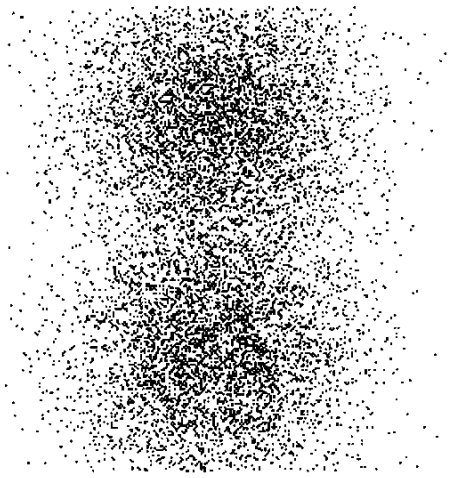


Fig. 2. Two bidimensional overlapping Gaussian distributions.

Using the FAM architecture [6], the authors reported an average Brier score of 0.984 using 1000 training patterns. The average number of ART_a categories is reported to be 8. For a Maxnode strategy, the system evolved to 20 categories and a Brier score of 0.979.

We trained the FAMR for this benchmark. The initial value for ρ_a was set to 0.7 and ρ_{ab} was set to 0. First, we used a constant relevance factor 1, and obtained the average Brier score 0.894, and an average number of 6.85 ART_a categories. It would be unfair to compare directly our results to the results in [6] since, in our experiments, the training set was processed on-line. In [6], the order dependence problem was alleviated by retraining the system on different permutations of the training set.

Second, we chose a relevance factor inversely proportional to the distance between the pattern and the line bisecting the segment of the two Gaussian centers. This way, we payed more attention to training patterns with high classification uncertainty from the overlapping area of the classes. The main idea is how to make use of additional knowledge (the Gaussian centers) in the learning phase. We did not obtained a significant improvement and we believe that a deeper investigation is necessary here. This problem is interesting because it is connected to learning in hybrid systems, where explicit rules are mixed with learning from examples.

E. Landsat satellite images

This part of the experiments was concerned with classification of Landsat satellite images as used in Statlog project [14]. The dataset can be obtained from UCI Repository of Machine Learning Databases and Domain Theories [15] and consists of subsections of a scene drawn from the original satellite images. The measurements comprise the intensities of four spectral bands from the same scene. Given these values, the purpose is to predict the target output of a pixel as belonging to one of the six classes. This is a challenging benchmark problem because of the noisy images. Each input pattern has 36 integer

TABLE III

PERFORMANCE FOR THE LANDSAT DATA. THE FAM RESULTS ARE THOSE REPORTED IN [17] AND THE PFAM RESULTS ARE FROM [16]. THE PFAM RESULTS ARE OBTAINED ON MULTIPLE NETWORKS.

Algorithm		$\overline{\rho}_a = 0.0$	$\overline{\rho}_a = 0.9$
FAM	Test set recognition rate(%)	83.0	89.0
	No. of ART_a Categories	89	704
PFAM	Test set recognition rate (%)	81.4	89.0
	No. of ART_a Categories	87	518
FAMR	Test set recognition rate (%)	81.45	87.5
	No. of ART_a Categories	40	340

value attributes. The training set contains 4435 samples and the test file has 2000 samples.

Lim and Harrison [16] used this dataset to compare PFAM's performance to that from [17]. In their off-line experiments training patterns were randomized to produce different ordering sets. Each set was used to train a different PFAM network. In the prediction mode, the results were averaged across five individual networks.

In order to test the FAMR's incremental learning ability, we did not use different orderings of the training set as in [16] and the original data was trained on a single network. Thus, we did not eliminate the order–dependency.

For values of $\overline{\rho}_a$ ($\overline{\rho}_a$ is the initial value for ρ_a) close to the ones used in [16], [17], the results (test set recognition rate, number of ART_a categories) are reported in Table III.

The results are rather good, compared to those from [16], taking into consideration that the decision of only one system was used. For instance, for $\overline{\rho}_a = 0$, the test set recognition rate was close to the one reported in [16], but for a smaller number of ART_a categories, and also for an incremental (not off-line) training.

The trade-off between a high recognition rate and a small number of ART_a categories is generally better in the case of FAMR than in the case of FAM and PFAM.

V. CONCLUSIONS AND FUTURE WORK

The Mapfield algorithm developed here expands the range of FAM applications by allowing us assignation of a relevance factor to each training pair. The FAMR probability estimation is computationally simple and converges with probability one to the posterior probability. When the initial relevance is zero and all other relevances are constant, FAMR is equivalent to PROBART.

Compared to the FAM probability estimator, FAMR shows similar or better performances with respect to the Brier score, test set recognition rate, and number of generated nodes. As a classifier, FAMR favorably compares with PFAM. The true benefits of using FAMR may come from using a relevance factor assigned to the training samples, improving the quality of the results, especially for probability estimation.

Usage of the mean square error (3) allows us to evaluate the rate of convergence. Choosing an adequate variable relevance factor can result in a faster convergence and a better performance of the network. This is left for further research work.

REFERENCES

[1] C. P. Lim and R. F. Harrison, "An incremental adaptive network for on-line supervized learning and probability estimation," *Neural Networks*, vol. 10, no. 5, pp. 925–939, 1997.

[2] ——, "ART-Based Autonomous Learning Systems: Part I - Architectures and Algorithms," in *Innovations in ART Neural Networks*, L. C. Jain, B. Lazzerini, and U. Halici, Eds. Springer, 2000.

[3] R. Polikar, L. Udpa, S. S. Udpa, and V. Honovar, "Learn++: An incremental learning algorithm for supervised neural networks," *IEEE Transactions on Systems, Man, and Cybernetics–PartC*, vol. 31, no. 4, pp. 497–508, 2001.

[4] G. A. Carpenter and S. Grossberg, "The ART of adaptive pattern recognition by a self-organizing neural network," *IEEE Computer*, vol. 21, no. 3, pp. 77–88, 1988.

[5] G. Carpenter, S. Grossberg, N. Markuzon, J. Reynolds, and D. Rosen, "Fuzzy ARTMAP: A neural network architecture for incremental supervised learning of analog multidimensional maps," *IEEE Transactions on Neural Networks*, vol. 3, no. 5, pp. 698–713, 1992.

[6] G. Carpenter, S.Grossberg, and J. Reynolds, "A fuzzy ARTMAP nonparametric probability estimator for nonstationary pattern recognition problems," *IEEE Transactions on Neural Networks*, vol. 6, no. 6, pp. 1330–1336, 1995.

[7] S. Marriott and R. F. Harrison, "A modified fuzzy ARTMAP architecture for the approximation of noisy mappings," *Neural Networks*, vol. 8, no. 4, pp. 619–641, 1995.

[8] D. Specht, "Probabilistic Neural Networks," *Neural Networks*, vol. 3, pp. 109–118, 1990.

[9] R. Andonie, "A converse H-theorem for inductive processes," *Computers and Artificial Intelligence*, vol. 9, pp. 159–167, 1990.

[10] R. Andonie and L. Sasu, "A Fuzzy ARTMAP Probability Estimator with Relevance Factor," in *Proceedings of the 11th European Symposium on Artificial Neural Networks (ESANN 2003)*, April 23-25, Bruges, Belgium, 2003.

[11] S. Gallant, *Neural Network Learning and Expert Systems*. MIT Press, 1994.

[12] V. Ciesielski, "Boundary points do not improve the accuracy of neural net classifiers," in *Proceedings of the Eights Australian Joint Conference on Artificial Intelligence*, Canberra, 1995, pp. 163–170.

[13] K. Lang and M. Witbrock, "Learning to tell two spirals appart," in *Proceedings 1988 Connectionist Models Summer School*, 1989, pp. 52–59.

[14] D. Michie, D. J. Spiegelhalter, and C. C. Taylor, *Machine Learning, Neural and Statistical Classification*. Oxford Press, 1994.

[15] K. Blacke, E. Keogh, and C. J. Merz. (1998) UCI Repository of Machine Learning Databases. [Online]. Available: http://www.ics.uci.edu/~learn/mlrepository.html

[16] C. P. Lim and R. F. Harrison, "ART-Based Autonomous Learning Systems: Part II - Applications," in *Innovations in ART Neural Networks*, L. C. Jain, B. Lazzerini, and U. Halici, Eds. Springer, 2000.

[17] R. Y. Asfour, "Fuzzy ARTMAP: Neural Networks for Multisensor Fusion and Classification," Ph.D. dissertation, Boston University, 1995.