MicroARTMAP: use of Mutual Information for Category Reduction in Fuzzy ARTMAP

E. Gómez Sánchez†, Y.A. Dimitriadis†, J.M. Cano Izquierdo‡, J. López Coronado‡
† Dept. de Teoría de la Señal y Comunicaciones e Ingeniería Telemática,
University of Valladolid,
Camino Viejo del Cementerio, s/n, 47011, Valladolid, Spain
e-mail: edugom@tel.uva.es, yannis@tel.uva.es
‡ Dept. de Automática y Electrónica Industrial,
University of Cartagena,
Paseo Alfonso XIII, 48, 30203 Cartagena, Murcia, Spain
e-mail: joscan@plc.um.es, coronado@plc.um.es

Abstract

A new architecture, called MicroARTMAP, is proposed to impact the category proliferation problem present in Fuzzy ARTMAP. It handles probabilistic information through the optimization of the mutual information between the input and output spaces, but allowing a small training error, thus avoiding overfitting. While reducing the number of categories used by Fuzzy ARTMAP, it holds several desirable properties, such as a correct treatment of exceptions and a fast algorithm, as opposed to other approaches like BARTMAP. In addition, it is shown that MicroARTMAP is less sensitive than Fuzzy ARTMAP with respect to the the pattern presentation order, and that it degrades less if the training set is noisy.

Keywords: Fuzzy ARTMAP, category proliferation, mutual information, exceptions

1 Introduction

Fuzzy ARTMAP [2] is a neural network architecture for conducting a supervised learning in a multidimensional framework. Fuzzy ARTMAP and other ART based networks share some properties that make them very suitable for applications requiring on-line performance. These properties include the solution of the stability-plasticity dilemma [5], which allows incremental learning on time-varying environments; fast stable learning, multiple generalization scales and fast convergence with a relatively small number of training patterns.

However, Fuzzy ARTMAP suffers from a category proliferation problem [3], i.e. the training algorithm recruits a large number of categories to represent the input space and its relations to the output space. Thus training produces a large rule set, often with redundant information, that will increase the test processing time. Furthermore, this set of rules may hardly be interpreted by a human supervisor, loosing one of the most interesting properties of neuro-fuzzy systems. In order to solve this problem, several approaches have been proposed. On one hand, changes in Fuzzy ARTMAP architecture have been introduced, as in dARTMAP [3], that may avoid category proliferation while preserving its on-line stable learning property. However, as shown in [7], dARTMAP is only successful under certain problem conditions. On the other hand, post-processing methods, like rule pruning [1], can be computationally costly and loose the on-line feature.

In addition, probabilistic information can be used to improve Fuzzy ARTMAP performance. In PROBART [6] the inter-ART map is replaced by a probabilistic map, suppressing the match tracking mechanism. Boosted ARTMAP (BARTMAP) [8] is proposed as a modification of PROBART, that performs an off-line evaluation of the training error after the on-line unsupervised clustering of the input space. If the prediction error on the training set is beyond a threshold, new training is performed with a higher vigilance parameter, i.e. creating finer categories. This approach pretends to optimize the size of categories so that its recruitment is reduced. However, exceptions cannot be handled appropriately, as shown later in this paper. Moreover, the algorithm can be computationally costly, since clustering of the input space is completely unsupervised.

In this paper, μARTMAP (read MicroARTMAP, use of Mutual Information for Category Reduction in ARTMAP) architecture is proposed, that combines probabilistic information in order to reduce the number of categories by optimizing their sizes, and the use of a match tracking mechanism that will allow the correct treatment of exceptions, and a fast training algorithm.

The rest of this paper is organized as follows: section 2 presents the new μARTMAP architecture and algorithm, pointing out the differences with Fuzzy ARTMAP, PROBART and BARTMAP. Due to space

constraints, the reader is supposed to be familiar with Fuzzy ARTMAP architecture and algorithm. Section 3 is devoted to an experimental comparison of these architectures and the proposed μARTMAP, through several synthetic benchmarks. Finally, section 4 draws the main conclusions.

2 μARTMAP

As stated above, BARTMAP [8] replaces the inter-ART map by a probabilistic map as in PROBART [6]. Furthermore, it endows each category with its own vigilance parameter ρ_i^a, instead of a common ρ^a applied to evaluate matching for all units in ARTa. All ρ_i^a are initialized to a (usually relaxed) value, and clustering of all patterns proceeds, while the probabilistic map will count relations between categories in ARTa and categories in ARTb. After all patterns have been presented, the total prediction error on the training set is calculated easily from the probabilistic map weights, and if this error ε is beyond a threshold ε_{max}, some units with high contribution to the error are deleted, the base ρ_i^a raised and the training patterns are presented again. However, this does not permit to treat exceptions correctly. To see this, consider task 4 in figure 1. Since BARTMAP has not a match tracking mechanism, categories *within* other categories will not be created, and therefore one category will code all patterns in the inner square, while *several* categories will be necessary to code the rest of the patterns. On the other hand, training Fuzzy ARTMAP with $\rho^a = 0$ and an adequate pattern presentation order, will produce only two categories: one for the whole outer square, and other for the *exception* inner square. Moreover, in Fuzzy ARTMAP will require a single training epochs while several will be necessary to achieve acceptable results with BARTMAP.

The proposed μARTMAP attempts to reduce the number of recruited neurons in Fuzzy ARTMAP by adaptively selecting ρ_i^a vigilance parameter for each category, and also maintain the training error under a threshold as BARTMAP does. However, it incorporates a reset mechanism that will allow to handle exceptions properly, and reduce the number of training epochs. To achieve this, the conditional entropy H between ARTb and ARTa categories is verified to be under a threshold H_{max} after the presentation of all patterns, but also the contribution h_i to H is compared after each pattern presentation to a threshold h_{max}, eventually firing the reset mechanism. The minimization of H is shown to be equivalent to the maximization of the mutual information between ARTa and ARTb categories, hence the name μARTMAP. The complete training algorithm is explained in detail in section 2.2.

2.1 Definitions

Given partitions of the input space I into sets I_i (not necessarily connected) and output space O into sets O_i, the conditional entropy $H(O \mid I)$, here denoted simply by H, is given by

$$H = \sum_i p_i \sum_j p_{ij} \log_2 p_{ij} \tag{1}$$

where p_i is the probability of occurrence of class I_i and p_{ij} is the joint probability of classes I_i and O_j. Let us denote

$$h_i = p_i \sum_j p_{ij} \log_2 p_{ij} \tag{2}$$

the contribution to H of set I_i.

It is important to remark that the mutual information (MI) between input and output spaces is given by $MI(O;I) = H(O) - H(O \mid I)$, where $H(O)$ is the entropy for the output space. Therefore, for a given $H(O)$ (as in classification tasks), minimizing the conditional entropy is equivalent to the maximization of the mutual information.

2.2 The architecture

The structure of μARTMAP architecture is similar that of Fuzzy ARTMAP, consisting of two unsupervised Fuzzy ART modules (ARTa and ARTb clustering the input and output spaces, respectively) and an associative map (inter-ART) storing relations between the unsupervised modules. The following changes are introduced:

	Fuzzy ARTMAP	**PROBART**	**BARTMAP**	**μARTMAP**
inter-ART	competitive map	probabilistic map		
vigilance	ρ^a for ARTa		ρ_i^a for each unit i in ARTa	
On-line reset for unit i in ARTa	if wrong prediction unit i is inhibited and ρ^a is raised	nothing		if $h_i > h_{max}$ unit i is inhibited
Off-line check	nothing		if $\varepsilon > \varepsilon_{max}$ unit with greater ε_i deleted and ρ_i^a raised for new units	if $H > H_{max}$ unit with greater h_i deleted and ρ_i^a raised for new units
Suitable for classification	Yes	No		Yes
Treatment of exceptions	Correct	Not correct		Correct
Control training error	No		Yes	

Table 1: Important features in Fuzzy ARTMAP, PROBART, BARTMAP and μARTMAP.

- Each unit in ARTa has its own vigilance parameter ρ_i^a, as in Boosted ARTMAP.
- The inter-ART map is a probabilistic map that stores joint probabilities p_{ij} (on-line) and P_{ij} (off-line), for units i in ARTa and j in ARTb. PROBART and BARTMAP also store p_{ij} weights.
- Two user-tuned parameters are introduced: h_{max} sets an upper limit to the contribution h_i (see eq. 2) of each unit i, calculated on-line; meanwhile, H_{max} sets an upper limit to the total entropy (see eq. 1), calculated off-line. In Boosted ARTMAP a user-tuned parameter ε_{max} imposes a limit to the final prediction error.

The algorithm for μARTMAP is as follows:

1. The probabilistic map is initialized with $p_{ij} = P_{ij} = 0$ for all i, j. The Fuzzy ART modules are initialized with $w_{ij}^a = w_{ij}^b = 1$. In addition $h_i = H_i = 0$ for all i.
2. **On-line stage:** For every pattern in the training set:
   (a) The pattern is presented, and according to Fuzzy ART algorithm [1] unit i wins in ARTa, while unit j wins in ARTb. If a new unit i is recruited in ARTa, its ρ_i^a is set to ρ^a.
   (b) The probability weights p_{ij} are tentatively updated.
   (c) The contribution to entropy of unit i is calculated, according to eq. (2), and compared to h_{max}. If
      - $h_i > h_{max}$, then the unit i in Fuzzy ARTa is inhibited, changes in step 2b are revoked and the pattern is presented again (step 2a).
      - $h_i \leq h_{max}$, then ARTa weights are updated according to [1], and the next pattern is processed.
3. **Off-line stage:** If $H_{max} \geq \log_2 N^b$, where N^b is the number of output classes, then this stage is not necessary, and therefore training keeps being on-line. Otherwise, for every pattern in the training set:
   (a) The pattern is presented, and according to Fuzzy ART algorithm [1] unit i' wins in ARTa, and unit j in ARTb. The i' could be different from the i unit selected for this training pattern in the on-line stage.
   (b) The values of $P_{i'}$ and $P_{i'j}$ are updated.
4. The values of h_i are calculated for all i, and also the value of H, using probabilities P_i and P_{ij}. If

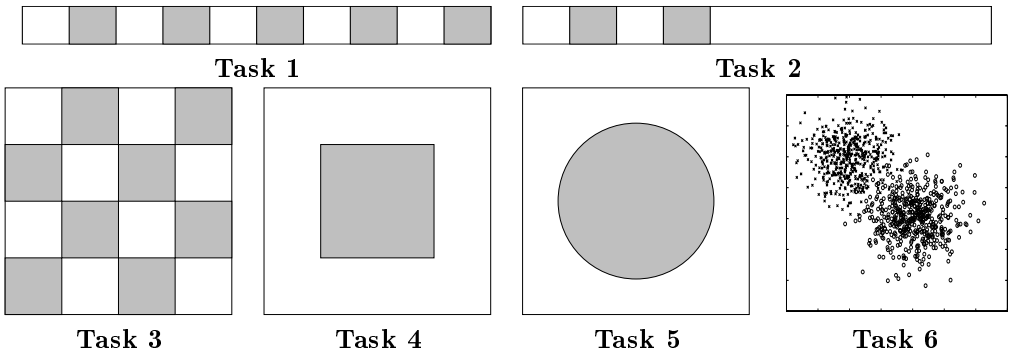

Figure 1: Synthetic benchmarks proposed to compare Fuzzy ARTMAP, BARTMAP and μARTMAP. Tasks 1 and 2 are one-dimensional problems, while tasks 3, 4, 5 and 6 are two-dimensional problems. In all cases there are two output classes.

- $H > H_{max}$, then the unit i with maximal contribution h_i to H is deleted. All the patterns classified in this unit are marked for newer presentation (again step 2), and the value of ρ^a is set by $\rho^a = \rho_i^a + \Delta\rho$.
- $H \leq H_{max}$ the training ends.

We can consider Fuzzy ARTMAP, PROBART and BARTMAP special cases of μARTMAP. If $h_{max} = 0$ and $H_{max} \geq \log_2 N^b$, then μARTMAP is similar to Fuzzy ARTMAP, except for the important fact that Fuzzy ARTMAP match tracking mechanism raises ρ^a temporarily, allowing the creation of finer categories [2]. The creation of finer categories is supplied in μARTMAP by the off-line stage, but unlike Fuzzy ARTMAP, it only is performed if necessary, as shown in section 3. In addition, if $h_{max}, H_{max} \geq \log_2 N^b$ then μARTMAP is totally equivalent to PROBART. Finally, if H_{max} is expressed as a function of ε_{max} and $h_{max} \geq \log_2 N^b$, then the proposed system is equivalent to BARTMAP. Table 1 summarizes the main features of the systems under study.

3 Experimental results

In order to evaluate the performance of the proposed architecture, several benchmarks are considered, as shown in figure 1. **Task 1** and **task 2** try to evaluate the optimization of the number of categories in one-dimensional problems. These tasks can be seen as identifying a large white class, with several gray exceptions. **Task 3** proposes a similar task in a two-dimensional space. **Task 4** shows a simple example of exception treatment that cannot be solved with two single categories by BARTMAP. **Task 5**, known as the *circle in the square*, is studied in [3], where distributed learning is proposed to reduce category proliferation in Fuzzy ARTMAP. Tasks 1 through 4 can be seen as different generalizations of the *circle in the square* problem, which is very used in ARTMAP literature [2, 3].

Finally **task 6** is proposed in the original BARTMAP paper [8], consisting of two normally distributed classes with means $(8, 12)$ and $(12, 8)$ respectively, and variance 2, in order to test how BARTMAP optimize the number of categories by allowing a small training error. The influence of noise in the training set is also studied in **task8**, consisting on the same classification problem in task 4, with Gaussian noise added to the input patterns.

For all the experiments shown in table 2, ten sets of 1000 training patterns were generated from an uniform distribution in the input space (except in task 6), while a single set of 10000 patterns was used for test. In task 6 input data were normalized to $[0, 1]$ as required by the Fuzzy ART modules of the three architectures. All processing times were measured in a 120MHz Pentium PC, with all the architectures implemented as `dlls` for MATLAB, using MATLAB's `clock` and `etime`.

Experimental results in table 2 show that μARTMAP performs optimally in the number of generated categories for tasks 1 and 2. Fuzzy ARTMAP is highly dependent on pattern presentation order to achieve

Task	Architecture	h_{max}	H_{max}	ε_{max}	$\Delta\rho$	N^a	t_{train}(s)	ε_{test}
Task 1	Fuzzy ARTMAP	—	—	—	—	13.2	0.07	0.73%
	BARTMAP	—	—	0.1	0.1	21.4	0.45	9.37%
	μARTMAP	0.0	0.1	—	0.1	**6.0**	0.38	**0.60%**
Task 2	Fuzzy ARTMAP	—	—	—	—	6.6	0.04	**0.41%**
	BARTMAP	—	—	0.1	0.1	5.9	0.37	8.74%
	μARTMAP	0.0	0.1	—	0.1	**3.0**	0.19	1.16%
Task 3	Fuzzy ARTMAP	—	—	—	—	37.3	0.31	8.05%
	BARTMAP	—	—	0.1	0.05	81.5	1.27	14.08%
	μARTMAP	0.0	0.1	—	0.05	**10.8**	2.14	**6.19%**
Task 4	Fuzzy ARTMAP	—	—	—	—	6.7	0.06	0.71%
	BARTMAP	—	—	0.1	0.05	12.0	0.44	11.86%
	μARTMAP	0.0	0.1	—	0.1	**2.0**	0.03	**0.30%**
Task 5	Fuzzy ARTMAP	—	—	—	—	24.3	0.27	**5.37%**
	BARTMAP	—	—	0.1	0.05	25.1	0.60	12.64%
	dARTMAP†	—	—	—	—	16.0	—	6.80%
	μARTMAP	0.0	0.2	—	0.1	**7.1**	0.40	6.16%
Task 6	Fuzzy ARTMAP	—	—	—	—	21.7	0.17	**4.16%**
	BARTMAP	—	—	0.25	0.1	**2.7**	0.67	11.03%
	μARTMAP	0.0	0.1	—	0.05	7.5	0.6	4.32%
Task 7	Fuzzy ARTMAP	—	—	—	—	16.7	0.17	2.79%
	BARTMAP	—	—	0.1	0.05	12.6	0.40	10.51%
	μARTMAP	0.1	0.2	—	0.05	**3.3**	0.18	**2.10%**

Table 2: Experimental results for the benchmarks shown in figure 1. Task 7 consists in the same classification problem of task 4 with Gaussian noise added to the input patterns. All experiments were carried out on ten different training sets of 1000 patterns, and the same test set of 10000 patterns (except †: results from [3]). In all networks $\rho^a = 0.0$ and $\beta^a = 1.0$.

optimal category recruitment [4], and therefore a larger number of categories is required in average. In addition, BARTMAP shows unsuitable for problems with *very probable exceptions*, due to the lack of an inter-ART reset mechanism. Training times are small for these problems, although it is clear that Fuzzy ARTMAP has the fastest algorithm. These results can be extended to two-dimensional cases, as shown by task 3. Here, μARTMAP does not reach an optimal solution in all the cases, since it also depends on the pattern presentation order as other ART based architectures.

In task 4, Fuzzy ARTMAP produces a number of redundant categories in the *white* region, due to the fact that it raises ρ^a after an inter-ART reset is produced. Instead, in μARTMAP ρ^a is raised in the off-line stage, and therefore it can be avoided when it is unnecessary, like in this task, where an optimal number of categories is found. Therefore, we can find as an emergent propierty that μARTMAP is less sensitive to pattern presentation order than Fuzzy ARTMAP. In this task, BARTMAP again fails to treat the single but very probable exception correctly, producing a large number of categories and a high test error.

Task 5, the *circle in the square* problem, is studied in [3], where distributed learning is proposed to reduce category commitment in Fuzzy ARTMAP. In [3] Distributed ARTMAP is reported to use 16 categories to produce 6.8% test error, when trained on a 1000 pattern training set. As seen in table 2, μARTMAP uses a reduced number of categories with higher test accuracy, by adequately positioning the categories, and allowing some error in the borders between classes. Both Fuzzy ARTMAP and BARTMAP used a much larger number of categories.

In [8] task 6 is proposed to show that BARTMAP can find a small set of categories that correctly classify points from two overlapping gaussians. The overlap region is not very probable, and since a small training error is allowed, just a few categories will be committed to code it. However, the experiments show that the test error is much higher than that of Fuzzy ARTMAP. μARTMAP also allows a small train entropy

(and therefore error), thus using a reduced number of categories, but achieving similar accuracy to Fuzzy ARTMAP. Fuzzy ARTMAP does not treat probabilistic information, and therefore will dedicate several categories to map the overlap. This result indicates that Fuzzy ARTMAP will degrade more in a noisy environment than BARTMAP or μARTMAP. To test this fact, task 7 is proposed, consisting in the same classification problem of task 4, with Gaussian noise ($\sigma = 0.01$) added to the input patterns. In table 2 it can be seen that the number of categories used by Fuzzy ARTMAP doubles, while BARTMAP and μARTMAP suffer a much smaller increase in complexity.

4 Conclusions

A new ARTMAP arcuitecture, called μARTMAP, has been proposed to impact the category proliferation problem present in Fuzzy ARTMAP. It handles propobabilistic information through the optimization of the mutual information between the clustering made on the input and output spaces. This optimization leads to a reduction in the training error, although a small error in the training set can be allowed, as in BARTMAP, thus avoiding data overfitting. However, whilst BARTMAP performs a totally off-line optimization of the training error, μARTMAP includes an on-line inter-ART reset mechanism, achieving fast convergence and correct treatment of populated exceptions.

After conducting the experiments it can be concluded that μARTMAP produces a much smaller number of categories than Fuzzy ARTMAP, at the cost of a higher processing time. In addtion, it has been seen that μARTMAP is less sensitive than Fuzzy ARTMAP with respect to the pattern presentation order. μARTMAP also outperforms BARTMAP in problems with well defined sollution that require the treatment of probable exceptions, while in problems where class overlapping exists with low probability μARTMAP performs similarly. Moreover, μARTMAP is computationally more efficient than BARTMAP.

In addition, it has been shown that in the realistic case of training data being corrupted by noise, Fuzzy ARTMAP degrades much more than BARTMAP and μARTMAP.

References

[1] G.A. Carpenter. Fuzzy ART. In B. Kosko, editor, *Fuzzy Engineering*. Prentice Hall, Carmel, 1994.

[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, September 1992.

[3] G.A. Carpenter, B.L. Milenova, and B.W. Noeske. Distributed ARTMAP: a neural network for fast distributed supervised learning. *Neural Networks*, 11:793–813, 1998.

[4] I. Dagher, M. Georgipoulos, G.L. Heileman, and G. Bebis. An ordering algorithm for pattern presentation in Fuzzy ARTMAP that tends to improve generalization performance. *IEEE Transactions on Neural Networks*, 10(4):768–778, July 1999.

[5] S. Grossberg. *Studies of mind and brain: Neural principles of learning, perception, development cognition, and motor control.* Reidel Press, Boston, MA, USA, 1982.

[6] S. Marriott and R. Harrison. A modified Fuzzy ARTMAP architecture for the approximation of noisy mappings. *Neural Networks*, 8(4):619–641, 1995.

[7] E. Parrado Hernández, E. Gómez Sánchez, Y.A. Dimitriadis, and J. López Coronado. A neuro-fuzzy system that uses distributed learning for compact rule set generation. In *Proceedings of the 1999 IEEE International Conference on System, Man and Cybernetics*, volume 3, pages 441–446, Tokyo, Japan, October 1999.

[8] S.J. Verzi, G.L. Heileman, M. Georgipoulos, and M.J. Healy. Boosted ARTMAP. In *Proc. of the IEEE World Congress on Computational Intelligence, WCCI'98*, pages 396–400, Anchorage, Alaska, May 1998.