PERGAMON

Neural Networks 14 (2001) 1279–1291

Neural Networks

www.elsevier.com/locate/neunet

Contributed article

Cross-validation in Fuzzy ARTMAP for large databases

Anna Koufakou, Michael Georgiopoulos*, George Anagnostopoulos, Takis Kasparis

School of Electrical Engineering and Computer Science, University of Central Florida, Engineering Building, Room 407, Orlando, FL 32816, USA

Received 4 December 2000; accepted 14 May 2001

Abstract

In this paper we are examining the issue of overtraining in Fuzzy ARTMAP. Over-training in Fuzzy ARTMAP manifests itself in two different ways: (a) it degrades the generalization performance of Fuzzy ARTMAP as training progresses; and (b) it creates unnecessarily large Fuzzy ARTMAP neural network architectures. In this work, we are demonstrating that overtraining happens in Fuzzy ARTMAP and we propose an old remedy for its cure: cross-validation. In our experiments, we compare the performance of Fuzzy ARTMAP that is trained (i) until the completion of training, (ii) for one epoch, and (iii) until its performance on a validation set is maximized. The experiments were performed on artificial and real databases. The conclusion derived from those experiments is that cross-validation is a useful procedure in Fuzzy ARTMAP, because it produces smaller Fuzzy ARTMAP architectures with improved generalization performance. The trade-off is that cross-validation introduces additional computational complexity in the training phase of Fuzzy ARTMAP. © 2001 Elsevier Science Ltd. All rights reserved.

Keywords: Fuzzy ARTMAP; Cross-validation; Overtraining; Generalization performance

1. Introduction

Fuzzy ARTMAP was introduced in the neural network literature by Carpenter, Grossberg, Markuzon, Reynolds and Rosen (1992a) and Carpenter, Grossberg and Rosen (1992) and since then it has been established as one of the premier neural network architectures in solving classification problems. At the same time that Fuzzy ARTMAP appeared in the neural network literature, other ART-like architectures have also been introduced with characteristics similar to Fuzzy ARTMAP (e.g. Healy, Caudell & Smith, 1993; Simpson, 1992, 1993). Furthermore, from Fuzzy ARTMAP's inception and until the present, a variety of researchers have proposed modifications of the Fuzzy ARTMAP neural network that have improved its performance (e.g. Carpenter & Markuzon, 1998; Carpenter & Ross, 1995; Charalampidis, Georgiopoulos & Kasparis, 2000; Williamson, 1996). Finally, ART architectures including Fuzzy ARTMAP and its modular pieces, such as Fuzzy ART, have been carefully analyzed in the literature by a multitude of investigators (e.g. Moore, 1988; Carpenter et al., 1992; Georgiopoulos, Dagher, Bebis & Heileman, 1999; Georgiopoulos, Fernlund, Bebis & Heileman, 1996; Georgiopoulos, Heileman & Huang, 1991; Georgiopoulos, Huang & Heileman, 1994; Huang, Georgiopoulos & Heileman, 1995).

In solving classification problems, Fuzzy ARTMAP has the capability of establishing arbitrary mappings between clusters of an input space of arbitrary dimensionality and clusters of an output space of arbitrary dimensionality. At times, in doing so it creates very large neural network architectures. As a result, a number of researchers have tried to address this problem with various degrees of success (e.g. Gomez Sanchez, Dimitriadis, Cano Izquierdo & Lopez Colorado, 2000; Vertzi, Heileman, Georgiopoulos & Healy, 1998; Williamson, 1996). In Vertzi et al. (1998), the authors discussed the issue of overtraining in Fuzzy ARTMAP. This issue is most apparent when the classes of the classification problem that Fuzzy ARTMAP tries to solve exhibit significant overlap and results in the creation of large Fuzzy ARTMAP neural network architectures. In this paper, we address the same problem, the problem of overtraining in Fuzzy ARTMAP. Overtraining in Fuzzy ARTMAP manifests itself in two different ways. It may decrease the generalization performance of the network or it may increase the size of the Fuzzy ARTMAP architecture (without necessarily improving its generalization), or both. To address the problem of overtraining in Fuzzy ARTMAP we propose the usage of cross-validation techniques. Cross-validation is a well respected procedure in the statistical literature that allows determination of when overtraining occurs. To avoid some of the issues that plague cross-validation approaches (e.g. the

* Corresponding author. Tel.: +1-407-823-5338; fax: +1-407-823-5835.
E-mail address: michaelg@mail.ucf.edu (M. Georgiopoulos).

0893-6080/01/$ - see front matter © 2001 Elsevier Science Ltd. All rights reserved.
PII: S0893-6080(01)00090-9

issue of small datasets) we focus our attention here only on databases that have a sufficient number of data points. This way, we can split the data into training, validation and test sets that are representative of the distribution that the data follow. In order to verify that the chosen training, validation, and test sets follow the actual distribution of the data we compared the histogram of a large set extracted from the data with the histogram of the chosen training, validation and test sets. The histograms of the chosen training, validation and test sets were a good match of the histogram produced by the large set extracted from the data.

There is a large and interesting literature on cross-validation methods which often emphasizes asymptotic statistical properties, or the calculation of generalization error for certain models. The literature is too large to survey here, so we restrict ourselves in a limited sample of papers that share some connection with the work conducted in this paper, and the foundational papers that include those of Stone (1974, 1977). In Kohavi (1995), three methods for accuracy estimation of a model and for model selection are discussed. The leave-one-out cross-validation, the *k*-fold cross-validation and the bootstrap method; the models considered include C4.5 and Naive Bayes. Kohavi's conclusion is that the best method is 10-fold cross-validation for accuracy estimation of a model and model selection. In our paper, we assume that we have enough data, and as a result we can claim that the correct data distribution is accurately represented by the training, validation or test sets. Consequently, we perform training of Fuzzy ARTMAP with a single training set, validation of Fuzzy ARTMAP with a single validation set and testing of Fuzzy ARTMAP with a single test set. Our experimental results indicate that we can trust this cross-validation approach in producing reliably good Fuzzy ARTMAP models. This method of performing cross-validation is also adopted by Amari, Murata, Muller, Finke and Yang (1996, 1997).

Another paper that is worth mentioning is the paper by Dietrich (1998). In this work the author discusses a taxonomy of statistical questions in machine learning, one of which is the selection of an appropriate pattern classifier under the assumption that the data available to us are plentiful. This is the same problem that we are focusing on here, from the perspective of which of a number of Fuzzy ARTMAP neural networks is the best classifier for the classification problem at hand. The type of Fuzzy ARTMAP networks investigated are (a) a Fuzzy ARTMAP network that is trained until completion, (b) a Fuzzy ARTMAP network that is trained for one epoch, and (c) a Fuzzy ARTMAP network that is trained to the point where its performance on the validation set is maximized.

Our review of the cross-validation is not complete if we do not mention some of the recent papers that have appeared in the literature and examine cross-validation procedures for another popular neural network architecture, the multi-layer perceptron (MLP) (see Rumelhart, Hinton & Williams, 1986). These papers include the work by Anders and Korn (1999) and Prechelt (1998). A lot more work on cross-validation to avoid overtraining in MLP has been reported in the literature The interested reader may want to consult the references in the two aforementioned recent publications. Our careful examination of the literature did not identify any references where Fuzzy ARTMAP training is stopped early through a cross-validatory procedure. As we have mentioned earlier, this is the topic that this paper addresses.

The organization of the paper is as follows. In Section 2 we provide the basic details of the Fuzzy ARTMAP architecture. In Section 3 we discuss the topic of cross-validation and explain what kind of cross-validation procedure is applied to Fuzzy ARTMAP training. In Section 4 we elaborate on the types of experiments conducted that compare Fuzzy ARTMAP networks that are (i) trained until completion, (ii) trained for one epoch, and (iii) trained until the maximum performance on the validation set is achieved. In Section 4 we focus on experiments with simulated data and real databases. The conclusions of our work are emphasized in the final section (Section 5) of this paper.

2. Fuzzy ARTMAP neural network architecture

The Fuzzy ARTMAP neural network consists of two Fuzzy ART modules, designated as ART_a and ART_b, as well as an inter-ART module as shown in Fig. 1. Inputs are presented at the ART_a module, while their corresponding outputs are presented at the ART_b module. The inter-ART module includes a MAP field whose purpose is to determine whether the correct mapping has been established from inputs to outputs.

Some pre-processing of the input patterns of the pattern classification task takes place before they are presented to the ART_a module of Fuzzy ARTMAP. The first pre-processing stages takes as an input as M_a-dimensional input pattern from the pattern classification task and transforms it into an output vector $\mathbf{a} = (a_1, \ldots, a_{M_a})$, whose every component lies in the interval [0, 1]. The second pre-processing stage accepts as an input the vector **a** of the first pre-processing stage and produces a vector **I**, such that

$$\mathbf{I} = (\mathbf{a}, \mathbf{a}^c) = (a_1, \ldots, a_{M_a}, a_1^c, \ldots, a_{M_a}^c) \tag{1}$$

where

$$a_i^c = 1 - a_i; 1 \leq i \leq M_a \tag{2}$$

The above transformation is called complement coding. Complement coding is performed in ART_a at a pre-processor field designated by F_0^a (see Fig. 1). From now on, we will be referring to the vector **I** as the input pattern. Similar type of operations, as the ones described above, are also performed in order to produce the output pattern **O** that is applied at the ART_b module.

Fuzzy ARTMAP frequently operates in two distinct phases: the training phase and the performance phase. The training phase of Fuzzy ARTMAP works as

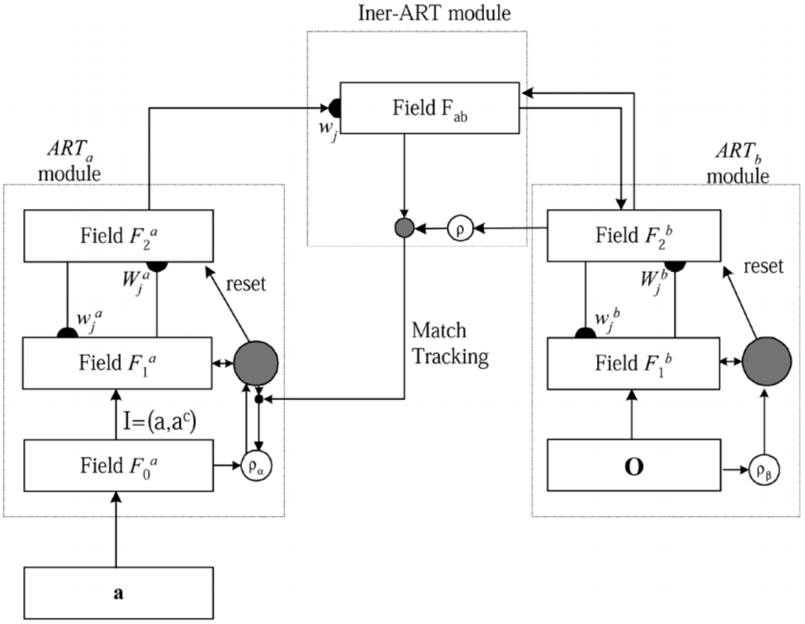

Fig. 1. Block diagram of the Fuzzy ARTMAP architecture.

follows: given a list of training input/output pairs, such as $\{\mathbf{I}^1, \mathbf{O}^1\}, \ldots, \{\mathbf{I}^r, \mathbf{O}^r\}, \ldots, \{\mathbf{I}^{NT}, \mathbf{O}^{NT}\}$, we want to train Fuzzy ARTMAP to map every input pattern of the training list to its own corresponding output pattern. In order to achieve the aforementioned goal, we present the training list repeatedly to the Fuzzy ARTMAP architecture. That is, present $\mathbf{I}^1$ to ART_a and $\mathbf{O}^1$ to ART_b, then $\mathbf{I}^2$ to ART_a and $\mathbf{O}^2$ to ART_b, and finally $\mathbf{I}^{NT}$ to ART_a and $\mathbf{O}^{NT}$ to ART_b; this corresponds to one list presentation. We present the training list as many times as it is necessary for Fuzzy ARTMAP to correctly classify all the input patterns. The task is considered accomplished (i.e. the learning is complete) when the weights do not change during a list presentation. The aforementioned training scenario is called off-line learning.

The performance phase of Fuzzy ARTMAP works as follows: given a list of test input patterns, such as $\tilde{\mathbf{I}}^1, \ldots, \tilde{\mathbf{I}}^r, \ldots, \tilde{\mathbf{I}}^{NS}$, we want to find the Fuzzy ARTMAP output produced when each one of the aforementioned test patterns is presented at its F_1^a field. In order to achieve the aforementioned goal, we present the test list once to the trained Fuzzy ARTMAP architecture.

More details about the training and performance phase of Fuzzy ARTMAP can be found in the literature (Carpenter, Grossberg & Reynolds, 1991). What is worth mentioning, though, is that for pattern classification problems, Fuzzy ARTMAP creates clusters of input data in the input pattern space. These clusters are hyperboxes that enclose within their boundaries all the input patterns that chose them as their representative clusters. At the end of training, the clusters (hyperboxes) created define appropriate decision regions that split the input space into subspaces that are mapped to a single output category (class). It is possible that more than one subspace of the input pattern space is mapped to the same output class.

There are a number of Fuzzy ARTMAP parameters that affect its performance in classification problems. These are: the choice parameter $\beta\ _a$, and the baseline vigilance parameter $\bar{\rho}_a$. The choice parameter assumes values in the interval (0, ∞), while the baseline vigilance parameter assumes values in the interval [0, 1]. The Fuzzy ARTMAP equations in which the choice parameter β_a, and the baseline vigilance parameter $\bar{\rho}_a$ appear are shown below.

$$T_j^a(\mathbf{I}) = \frac{|\mathbf{I} \wedge \mathbf{w}_j^a|}{\beta_a + |\mathbf{w}_j^a|} \tag{3}$$

$$\frac{|\mathbf{I} \wedge \mathbf{w}_j^a|}{\mathbf{I}} \geq \rho_a \tag{4}$$

The first equation above calculates the bottom-up input applied at node j of F_2^a (choice function) due to the presentation of pattern $\mathbf{I}$ at the field F_1^a of the Fuzzy ARTMAP architecture. The node in F_2^a that receives the maximum bottom-up input is chosen to represent the input pattern $\mathbf{I}$. The node thus chosen in F_2^a is considered appropriate to represent the input pattern $\mathbf{I}$ if and only if it satisfies inequality (4), which is referred to as the vigilance criterion. The right hand side of inequality (4) is the vigilance parameter ρ_a in ART_a that is initially set equal to the baseline vigilance $\bar{\rho}_a$. During Fuzzy ARTMAP training, the vigilance parameter value is allowed to increase above the baseline vigilance parameter value; the range of the vigilance parameter is the interval $[\bar{\rho}_a, 1]$. Small values of this baseline vigilance parameter result in coarser clustering of the input patterns, while large values of the baseline vigilance result in finer clustering of the input patterns of the pattern classification task. The choice parameter β_a has an effect on the order according to which nodes in F_2^a will be accessed due to the presentation of an input pattern applied at F_1^a field of Fuzzy ARTMAP (for more details see Georgiopoulos et al., 1999, 1996).

Another Fuzzy ARTMAP parameter that is often not referred to as a parameter in the associated literature is the order of training presentation. It has already been an established fact that Fuzzy ARTMAP performance depends on the order according to which data are presented to Fuzzy ARTMAP during the training process. As a result, Fuzzy ARTMAP's performance is frequently evaluated by averaging the performance of Fuzzy ARTMAP for different orders of training data presentation to it. In this paper, the values of β_a and ρ_a parameters in ART_a were chosen equal to 0.01 and 0, respectively; this choice of ART_a parameter values guarantees that the network architectures that Fuzzy ARTMAP creates are small (compared to larger values for β_a and ρ_a).

3. Cross-validation

Estimating the accuracy of a classifier induced by supervised learning methods, such as Fuzzy ARTMAP, is an important issue. One of the reasons for its importance is that it gives us some guidance on how good the future predictive accuracy of the classifier is. Another, equally important reason, is that it gives us a way of choosing the 'best' classifier model amongst a set of classifier models.

Cross-validation is a statistical technique that allows us to estimate the accuracy of a classifier model. Kohavi (1995) discusses two prominent cross-validation procedures. The first one is referred to as the hold-out method. We split the set S of available data into a training set S_{tr} and a validation set S_{v}. The classifier is designed using the data in the training set S_{tr} and its accuracy is estimated by evaluating its performance on the validation set S_{v}. That is, the hold-out estimated accuracy is defined as

$$PCC_{\mathrm{v}} = 100 \times \frac{1}{NV} \sum_{(I_i,O_i)\in S_{\mathrm{v}}} \delta(y_i, O_i) \tag{5}$$

where PCC_{v} denotes the percentage of correct classification of the classifier over the validation set S_{v}, NV is the number of datapoints in validation set S_{v}, the I_i and O_i designate the i-th input and desired output pair in S_{v}, y_i is the actual response of the classifier when it is excited by the input I_i and $\delta(x, y) = 1$ if $x = y$, while $\delta(x, y) = 0$ if $x \neq y$.

Obviously, the hold-out estimate is a random number that depends on the division of the available data in S into a training set S_{tr} and a validation set S_{v}. Often the hold-out method is repeated k times and the estimated accuracy PCC_{v} is produced by averaging the estimated accuracies of the k runs.

The second method for cross-validation is referred to as k-fold cross-validation. In this procedure, the available data S are split into k mutually exclusive subsets, designed as S^1, S^2, …, S^k of approximately equal size. The classifier is trained and tested (validated) k times. Each time m, $\mathrm{m} \in \{1, 2, \ldots, k\}$, it is trained on $S \backslash S^m$ and tested on S^m. The cross-validation estimate is defined as the number of correct classifications divided by the number of data points in the set S. That is,

$$PCC_v = 100 \times \frac{1}{NV} \sum_{m=1}^{k} \sum_{(I_i,O_i)\in S^m} \delta(O_i, y_i) \tag{6}$$

where PCC_{v} is the percentage of correct classification on the validation set (which in this case happens to be the entire set of available data), NV is the number of elements in S_{v} (which happens to be the same as S), (I_i, O_i) represents a generic input/desired output pair in S^m, and y_i is the actual output of the classifier, designed with data in $S \backslash S^m$, and excited with the input I_i from the set S^m. Once more, $\delta(x, y) = 1$ if $x = y$, while $\delta(x, y) = 0$ if $x \neq y$.

Obviously the cross-validation estimate in Eq. (6) is a random number that depends on the division into folds. Complete cross-validation is the average of the above estimates over all the possible folds of NT training data into k folds of approximately equal size. This is too expensive though, except in the case of 1-fold cross-validation, with NT relatively small. As Kohavi states, repeating cross-validation multiple times using different splits into folds provides a better estimate at the expense of additional computational cost. In stratified cross-validation, the folds are stratified so that they contain approximately the same proportions of labels as the original set.

In this paper, we use stratified cross-validation to stop training of Fuzzy ARTMAP at a point where its performance on the validation set is maximized. To produce the estimate of the Fuzzy ARTMAP performance we used the hold-out cross-validation technique. Since we are focusing on datasets with large samples of data we do not have to worry about making inefficient use of the available data. Furthermore, since we deal with large databases we did not use k-fold cross-validation to avoid increased computational costs. Despite the fact that one of the major advantages of Fuzzy ARTMAP is that it is an instance-based classifier, its performance on a number of databases where it is trained off-line has also been investigated (e.g. Carpenter & Markuzon, 1998; Carpenter et al., 1992a). Some of these databases, such as the Letters database in Carpenter et al. (1992), and others, could very well be thought of as large databases.

4. Experiments—Results—Observations

We conducted two sets of classification experiments to demonstrate the potential of cross-validation in Fuzzy ARTMAP. The first set of experiments dealt with artificial databases, and the second set of experiments dealt with real databases. The advantage of using artificial databases is that we can generate as many training, validation, and test data as we want. Hence, we can easily satisfy the assumption in this paper that we are dealing with large databases. The other advantage of the artificial databases is that we can experiment with different values for the dimensionality of the input patterns, the number of output classes, and the degree of overlap of data belonging to different classes. The degree of overlap of data belonging to different classes affects the difficulty of the problem under consideration. Obviously, the problem becomes more difficult as the degree of overlap increases.

4.1. Artificial databases

The artificial databases consist of Gaussian data that are of dimensionality 2 or 5 or 10. They belong to either two different classes or three different classes. The degree of overlap of data that belong to different classes is either low, medium, or high. The Gaussian data generated are independent in different dimensions and their means and variances are chosen appropriately so that they can justify the characterization of low, medium, or high overlap.

For example, let us assume that we have a collection of Gaussianly distributed data, of dimensionality 2, that belong to two different classes. We decided to use 5000 datapoints per class to train Fuzzy ARTMAP (this set is S_{tr}), 5000 different datapoints per class to cross-validate Fuzzy ARTMAP (this set is S_{v}), and 5000 different datapoints per class to test the performance of the trained Fuzzy ARTMAP (this set is S_{tes}). We trained Fuzzy ARTMAP in three different modes:

Mode 1: Train Fuzzy ARTMAP with the training data until completion (i.e. until Fuzzy ARTMAP's misclassification rate on the training data is 0%). Evaluate the performance of the trained Fuzzy ARTMAP on the test data (S_{tes}). This performance is denoted by $PCC^{\mathrm{c}}_{\mathrm{tes}}$.

Mode 2: Train Fuzzy ARTMAP for one complete epoch (an epoch of training corresponds to one presentation of all input/output pairs of the training set through Fuzzy ARTMAP). Evaluate the performance of the trained Fuzzy ARTMAP on the test data (set S_{tes}). This performance is denoted by $PCC^{\mathrm{1EP}}_{\mathrm{tes}}$.

Mode 3: Train Fuzzy ARTMAP for one complete epoch but check its performance on the validation set (set S_{v}) every 100 iterations of training (an iteration of training corresponds to one input/output training pair presentation to Fuzzy ARTMAP). At the end of the one epoch of training we identify the iteration number at which the trained Fuzzy ARTMAP has exhibited the maximum performance on the validation set. We denote this performance as PCC_{v}. The weights of the Fuzzy ARTMAP that exhibited the maximum performance on the validation set are retained. These weights are then used to evaluate Fuzzy ARTMAP's performance on the test set (set S_{tes}). We denote this performance by PCC_{tes}.

For all the aforementioned three modes of training, we also retained the information about the number of nodes that the trained Fuzzy ARTMAP had created. We denoted the number of these nodes as N^{c}_{a}, N^{1EP}_{a}, and N_a, for modes 1, 2 and 3 of training, respectively. For the artificial databases mode 3 cross-validation was performed only for the first epoch of training, due to the fact that cross-validation is a computationally expensive procedure. We observed that for the artificial databases, performing cross-validation only for the first epoch of training was enough, since we were able to produce a small Fuzzy ARTMAP architecture with a good generalization performance.

Our experimental results with the artificial databases are illustrated in three different tables (Tables 1–3). In Table 1 we depict the results in seven different columns. Column 1, designated as Overlap, defines the degree of overlap between the data belonging to different classes. As we have emphasized above, the overlap degree levels that we are investigating are low, medium and high. In Fig. 2 we show Gaussianly distributed data belonging to two different classes that are of low overlap (see Fig. 2(a)), medium overlap (see Fig. 2(b)) and high overlap (see Fig. 2(c)). For example, in Fig. 2(a) the Gaussian data of class 1 have a mean vector of $(0, 0)^{\mathrm{T}}$, and variance vector $(1, 1)^{\mathrm{T}}$, while the data of class 2 have a mean vector of $(3.2, 3.2)^{\mathrm{T}}$, and variance vector $(1, 1)^{\mathrm{T}}$. In a similar fashion, in Fig. 2(b) the Gaussian data of class 1 have a mean vector of $(0, 0)^{\mathrm{T}}$, and variance vector $(1, 1)^{\mathrm{T}}$, while the data of class 2 have a mean vector of $(1, 1)^{\mathrm{T}}$, and variance vector $(1, 1)^{\mathrm{T}}$. Finally, in Fig. 2(c) the Gaussian data of class 1 have a mean vector of $(0, 0)^{\mathrm{T}}$ and variance vector $(1, 1)^{\mathrm{I}}$, while the data of class 2 have a mean vector of $(3.2, 3.2)^{\mathrm{T}}$, and variance vector $(4, 4)^{\mathrm{T}}$. The second column of Table 1 depicts the number of classes in our dataset; as we have mentioned before we have experimented with data belonging to two or three distinct classes. The third column in Table 1 shows the dimensionality of the input patterns. As we have said before, we have experimented with data of dimensionality 2, 5 and 10.

To discuss the rest of the columns of Table 1 and Tables 2 and 3, let us focus on one of the rows of Table 1, the boldfaced entry of the medium overlap category corresponding to data of dimensionality 10, belonging to three classes. The results reported in columns 4–8 of the boldfaced entry of the medium overlap category are extracted by averaging the results over 25 experiments. These experiments were constructed by taking five different sets of training/validation/test data and for each such set of data we trained Fuzzy ARTMAP with five distinct orders of training data presentations (chosen randomly). For future reference, we refer to these five different sets of data as S^{m}_{tr}, S^{m}_{v}, and S^{m}_{tes}, for $1 \leq m \leq 5$. For each one of these sets, we refer to the five orders of training data presentation by $or(m)$, where $or(m)$ takes the values 1, 2, 3, 4, 5 to designate the five different orders of presentation for each one of the five training data sets. The entry of the fourth column of the boldfaced row in the medium overlap category corresponds to $\overline{\overline{PCC^{\mathrm{c}}_{\mathrm{tes}}}}$. The

Table 1
Comparison of average percentage of correct classification (*PCC*s) and average node compression ratios (*CR*s) for the three different Fuzzy ARTMAP training modes (1, 2, 3) and three degrees of overlap (low, medium, high) using artificial databases

Overlap	Classes	Dim.	$\overline{\overline{PCC^{c}_{tes}}}$	$\overline{\overline{PCC_{tes}}} - \overline{\overline{PCC^{c}_{tes}}}$	$\overline{\overline{PCC_{tes}}} - \overline{\overline{PCC^{1EP}_{tes}}}$	$\overline{\overline{CR^{c}}}$	$\overline{\overline{CR^{1EP}}}$
Low	2	2	95.39	1.11	1.82	44.21	14.94
	2	5	96.78	0.79	1.92	19.31	5.04
	2	10	99.76	0.08	0.43	3.26	2.02
	3	2	99.95	0.86	1.51	45.32	15.75
	3	**5**	**99.19**	**0.08**	**0.51**	**10.31**	**3.82**
	3	10	99.57	0.31	0.68	3.23	2.05
Medium	2	2	84.50	2.69	4.34	63.54	23.34
	2	5	83.03	0.29	2.44	42.98	10.87
	2	10	83.59	1.27	3.66	18.38	4.27
	3	2	85.22	2.31	4.20	75.19	28.55
	3	5	83.51	2.61	4.81	55.84	14.34
	3	**10**	**85.66**	**2.38**	**4.34**	**34.75**	**7.91**
High	2	2	70.34	2.53	3.96	44.97	18.22
	2	5	68.09	2.43	3.94	51.45	14.17
	2	10	68.05	2.73	4.24	28.97	6.89
	3	**2**	**67.22**	**3.02**	**4.95**	**91.00**	**43.71**
	3	5	63.61	2.24	3.90	93.10	28.90
	3	10	73.06	1.01	2.62	17.96	4.14

Table 2
Comparison of average percentage of correct classification (*PCC*s) and average node compression ratios (*CR*s) for the three different Fuzzy ARTMAP training modes (1, 2, 3) and three degrees of overlap (low, medium, high) using artificial databases. The underscore in some entries indicates that the corresponding entry is a vector of an appropriate dimensionality

Class 1		Class 2		Class 3		Percentage of correct classification		Nodes	
Var	Mean	Var	Mean	Var	Mean	$\overline{PCC_{tes}} - \overline{PCC^{c}_{tes}}$	$\overline{PCC_{tes}} - \overline{PCC^{1EP}_{tes}}$	$\overline{CR^{c}}$	$\overline{CR^{1EP}}$
(a) Low overlap (3 classes, 5 dimensions)									
$\underline{1}$	$\underline{1}$	$\underline{1}$	$\underline{2}$	$\underline{1}$	$\underline{7}$	− 0.3	0.88	21.13	6.46
$\underline{1}$	$\underline{0}$	$\underline{1}$	$\underline{3}$	$\underline{1}$	6	0.3	0.7	8.87	3.80
$\underline{\mathbf{1}}$	$\underline{\mathbf{0}}$	$\underline{\mathbf{2}}$	$\underline{\mathbf{4}}$	$\underline{\mathbf{2}}$	$\underline{\mathbf{10}}$	**0.1**	**0.14**	**2.67**	**1.87**
$\underline{1}$	$\underline{0}$	$\underline{2}$	$\underline{6}$	$\underline{2}$	$\underline{12}$	0.01	0.01	1.33	1.33vr
$\underline{1}$	$\underline{0}$	$\underline{4}$	$\underline{5}$	$\underline{4}$	$\underline{10}$	0.28	0.81	17.56	5.63
(b) Medium overlap (3 classes, 10 dimensions)									
$\underline{1}$	$\underline{1}$	$\underline{1}$	2, 2, $\underline{0}$	$\underline{1}$	5, 5, $\underline{1}$	2.45	4.50	41.74	8.76
$\underline{1}$	$\underline{0}$	$\underline{1}$	2, 2, $\underline{0}$	$\underline{1}$	3, 3, $\underline{5}$	3.11	4.05	37.40	9.48
$\underline{\mathbf{1}}$	$\underline{\mathbf{0}}$	$\underline{\mathbf{2}}$	**3, 3,** $\underline{\mathbf{0}}$	$\underline{\mathbf{2}}$	**6, 6,** $\underline{\mathbf{1}}$	**3.03**	**5.40**	**45.33**	**10.25**
$\underline{1}$	$\underline{0}$	2	3, 3, 0	$\underline{2}$	7, 7, $\underline{0}$	3.54	5.25	26.41	5.96
$\underline{1}$	$\underline{0}$	$\underline{4}$	4, 4, $\underline{0}$	$\underline{14}$	8, 8, $\underline{1}$	− 0.25	2.51	22.86	5.13
(c) High overlap (3 classes, 2 dimensions)									
$\underline{1}$	$\underline{0}$	$\underline{1}$	1, 2	$\underline{1}$	2.5, 1.5	4.69	7.56	87.57	44.75
$\underline{1}$	$\underline{0}$	$\underline{1}$	2, 3	$\underline{1}$	1, 0.5	2.12	3.94	90.03	44.41
$\underline{\mathbf{1}}$	$\underline{\mathbf{0}}$	$\underline{\mathbf{2}}$	$\underline{\mathbf{2}}$	$\underline{\mathbf{2.5}}$	$\underline{\mathbf{3.6}}$	**4.4**	**6.26**	**167.28**	**79.35**
$\underline{1}$	$\underline{0}$	$\underline{4}$	$\underline{2}$	$\underline{4}$	$\underline{4}$	2.16	3.76	76.16	36.09
$\underline{1}$	$\underline{0}$	$\underline{4}$	$\underline{3}$	$\underline{4}$	$\underline{5}$	1.75	3.25	33.96	13.96

entry of the fifth column of the boldfaced row in the medium overlap category corresponds to $\overline{\overline{PCC_{tes}}} - \overline{\overline{PCC^{c}_{tes}}}$. The quantities $\overline{\overline{PCC_{tes}}}$ and $\overline{\overline{PCC^{c}_{tes}}}$ are defined as follows:

$$\overline{\overline{PCC_{tes}}} = \frac{1}{25}\sum_{m=1}^{5}\sum_{or(m)=1}^{5} PCC_{tes}(m, or(m)) \tag{7}$$

$$\overline{\overline{PCC^{c}_{tes}}} = \frac{1}{25}\sum_{m=1}^{5}\sum_{or(m)=1}^{5} PCC^{c}_{tes}(m, or(m)) \tag{8}$$

where $PCC_{tes}(m,\ or(m))$ is the performance of Fuzzy ARTMAP on the test data S^{m}_{tes}, trained under mode 3, with training data S^{m}_{tr} presented to it in the order $or(m)$, while $PCC^{c}_{tes}(m,\ or(m))$ is the performance of Fuzzy ARTMAP on the test data S^{m}_{tes}, trained under mode 1, with training data S^{m}_{tr} presented to it in the order $or(m)$.

Note that the entries of the fourth column of Table 1, which correspond to the average percentage of correct classification for mode 1 Fuzzy ARTMAP (complete training scenario) are a quantitative verification that we are dealing

Table 3
Comparison of percentage of correct classification (*PCC*s) and number of nodes created for the three different Fuzzy ARTMAP training modes (1, 2, 3) and three degrees of overlap (low, medium, high) using artificial databases

Order	PCC_v	PCC_{tes}	PCC^c_{tes}	PCC^{1EP}_{tes}	PCC_{tr}	PCC^c_{tr}	PCC^{1EP}_{tr}	N_a	N^c_a	N^{1EP}_a
(a) Low overlap case										
1	99.96	99.87	99.84	99.77	99.92	100	99.91	3	8	5
2	99.97	99.95	99.85	99.8	99.95	100	99.93	3	8	5
3	99.97	99.97	99.85	99.82	99.85	100	99.94	3	8	6
4	99.97	99.97	99.84	99.83	99.93	100	99.93	3	8	6
5	99.96	99.97	99.85	99.81	99.94	100	99.97	3	8	6
(b) Medium overlap case										
1	84.19	83.68	80.57	78.77	87.67	100	86.67	9	323	69
2	81.86	81.63	79.81	76.87	86.25	100	87.11	7	336	73
3	82.87	83.32	80.53	76.25	87.15	100	84.1	6	316	72
4	85.13	85.44	81	78.51	90.29	100	88.05	7	315	75
5	82.52	82.73	79.76	79.39	89.39	100	87.59	7	342	80
(c) High overlap case										
1	72.71	73.59	65.8	63.41	74.45	100	80.59	12	3043	1425
2	68.26	68.89	65.07	64.29	68.77	100	82.09	28	3182	1544
3	66.92	67.13	65.77	63.82	67.25	100	79.93	20	3046	1394
4	64.81	69.91	65.98	63.99	71.66	100	80.99	18	3061	1464
5	68.04	70.36	65.27	63.09	70.51	100	80.32	16	3093	1439

with a low, medium or high overlap. The $\overline{\overline{PCC^c_{tes}}}$ value for the low overlap is in the high 90s range, the medium overlap is in the low to mid-80s range and the high overlap is in the 60–70s range. The entry of the sixth column of the boldfaced row in the medium overlap category corresponds to $\overline{\overline{PCC_{tes}}} - \overline{\overline{PCC^{1EP}_{tes}}}$, which is the average difference in the percentage of correct classification between the mode 3 and mode 2 trained Fuzzy ARTMAPs. The seventh column, designated as $\overline{\overline{CR^c}}$, corresponds to the average ratio of the number of nodes created by the mode 1 trained Fuzzy ARTMAP and the number of nodes created by the mode 3 trained Fuzzy ARTMAP. This ratio is referred to as compression ratio complete (CR^c), to remind us how much mode 3 trained Fuzzy ARTMAP compresses the information compared to mode 1 trained Fuzzy ARTMAP (which is trained to completion). The eighth column, designated as $\overline{\overline{CR^{1EP}}}$, corresponds to the average ratio of the number of nodes created by the mode 2 trained Fuzzy ARTMAP and the number of nodes created by the mode 3 trained Fuzzy ARTMAP. This ratio is referred to as compression ratio one epoch (CR^{1EP}), to remind us how much mode 3 trained Fuzzy ARTMAP compresses the information compared to mode 2 trained Fuzzy ARTMAP (which is trained for one epoch). The definitions of the quantities $\overline{\overline{PCC^{1EP}_{tes}}}$, $\overline{\overline{CR^c}}$, and $\overline{\overline{CR^{1EP}}}$ are similar with the definitions of the quantities $\overline{\overline{PCC_{tes}}}$ and $\overline{\overline{PCC^c_{tes}}}$, defined in Eqs. (7) and (8).

The purpose of Table 2 is to take three entries from Table 1 (boldfaced one of low overlap, boldfaced one of medium overlap and boldfaced one of high overlap) and expand them in a way that provides more detailed information about these entries. For example, the boldfaced entry of medium overlap of Table 1 has been appropriately expanded in Table 2(b). The first row of Table 2(b) corresponds to a specific training/validation/test sets (i.e. S^1_{tr}/ S^1_v/ S^1_{tes}). The second row of Table 2(b) corresponds to a specific training/validation/test sets (i.e. S^2_{tr}/ S^2_v/ S^2_{tes}), and so forth for the rest of the rows of Table 2(b). Each row of Table 2(b) has 10 columns. The first six columns give us information about the statistics of the data involved. In particular, the first six entries of the boldfaced row of Table 2(b) tell us that class 1 data have mean vector $\underline{0}$, and variance vector $\underline{1}$, class 2 data have mean vector $(3, 3, \underline{0})^T$, and variance $\underline{2}$, and class 3 data have mean vector $(6, 6, \underline{1})^T$, and variance vector $\underline{1}$. The remaining four columns of the boldfaced entry in Table 2(b) (row 3) have the same interpretation as the last four columns in Table 1. The only difference is that an entry in Table 2(b) corresponds to an average of five experiments, while an entry in Table 1 corresponds to an average of 25 experiments. In particular, the entry of the seventh column of the boldfaced row of Table 2(b) corresponds to $\overline{PCC_{tes}} - \overline{PCC^c_{tes}}$ is defined as follows:

$$\overline{PCC_{tes}} = \frac{1}{5} \sum_{or(3)=1}^{5} PCC_{tes}(3, or(3)) \tag{9}$$

$$\overline{PCC^c_{tes}} = \frac{1}{5} \sum_{or(3)=1}^{5} PCC^c_{tes}(3, or(3)) \tag{10}$$

where $PCC_{tes}(3,\ or(3))$ is the performance of Fuzzy ARTMAP on the test data S^3_{tes}, trained under mode 3, with training data S^3_{tr} presented to it in the order $or(3)$, while $PCC^c_{tes}(3,\ or(3))$ is the performance of Fuzzy ARTMAP on the test data S^3_{tes}, trained under mode 1, with training data S^3_{tr} presented to it in the order $or(3)$. Similar interpretations are valid for the rest of the columns of row 3 of Table 2(b)

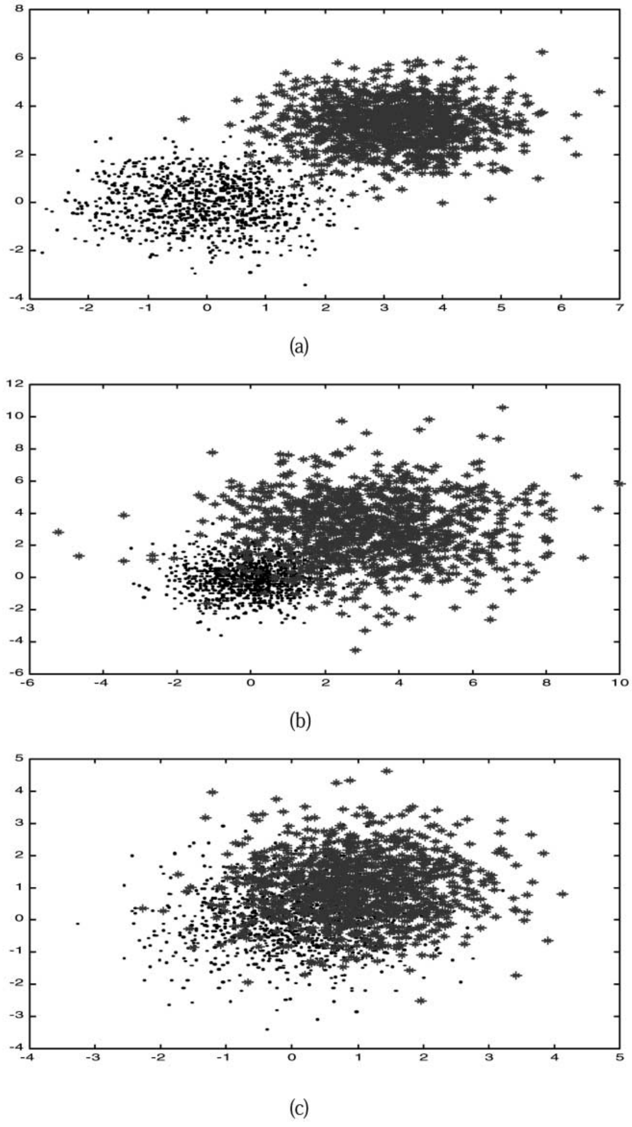

Fig. 2. Scatter plots of two-dimensional Gaussian classes for different class overlap degrees: (a) low overlap; (b) medium overlap and (c) high overlap.

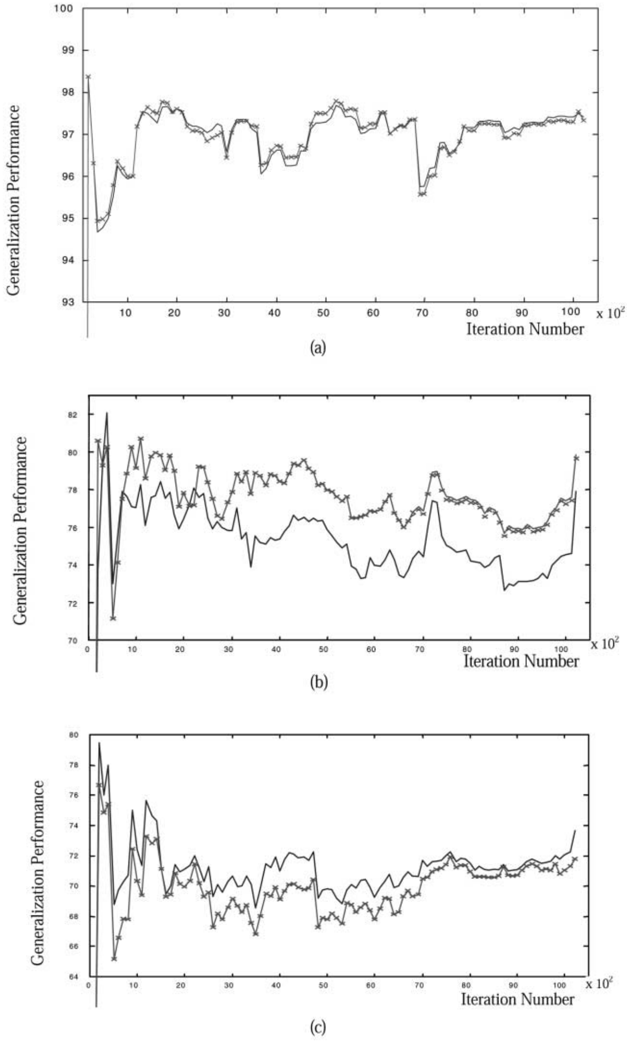

Fig. 3. Percentage of correct classification on the validation set (–) and testing set (×) of a trained Fuzzy ARTMAP network with data of different degrees of overlap (artificial databases): (a) low overlap; (b) medium overlap and (c) high overlap.

(i.e. columns 8, 9 and 10), as well as the rest of the rows of Table 2(b) and the entries in Table 2(a) and Table 2(c).

The purpose of Table 3 is to take three entries of Table 2 (boldfaced one from Table 2(a), boldfaced one from Table 2(b) and boldfaced one from Table 2(c)) and expand them, in a way that provides more detailed information about these entries. For example, Table 3(b) takes the third entry of Table 2(b) (corresponding to training set/validation set/test set equal to S_{tr}^3, S_{v}^3, and S_{tes}^3) and provides the values of $PCC_{\mathrm{v}}(3, or(3))$, $PCC_{\mathrm{tes}}(3, or(3))$, $PCC_{\mathrm{tes}}^{\mathrm{c}}(3, or(3))$, $PCC_{\mathrm{tes}}^{\mathrm{1EP}}(3, or(3))$, $PCC_{\mathrm{tr}}(3, or(3))$, $PCC_{\mathrm{tr}}^{\mathrm{c}}(3, or(3))$, $PCC_{\mathrm{tr}}^{\mathrm{1EP}}(3, or(3))$, $N_a(3, or(3))$, $N_a^{\mathrm{c}}(3, or(3))$, and $N_a^{\mathrm{1EP}}(3, or(3))$, for a Fuzzy ARTMAP that is trained with the data in S_{tr}^3, according to order of presentation $or(3) = 1, 2, 3, 4, 5$, validated with data in S_{v}^3, and tested with data S_{tes}^3. For conciseness of notation in Table 3(b) the above quantities are denoted as PCC_{v}, PCC_{tes}, $PCC_{\mathrm{tes}}^{\mathrm{c}}$, $PCC_{\mathrm{tes}}^{\mathrm{1EP}}$, PCC_{tr}, $PCC_{\mathrm{tr}}^{\mathrm{c}}$, $PCC_{\mathrm{tr}}^{\mathrm{1EP}}$, N_a, N_a^{c}, and N_a^{1EP}. Similar explanations are valid for the rest of the entries in Table 3(b) and the entries in Tables 3(a) and 3(c).

In Fig. 3 we illustrate the validation set performance and the test set performance of Fuzzy ARTMAP trained under mode 3 for three different overlap categories (low, medium and high). The curves in Fig. 3(a) correspond to a specific training/validation/test sets and a specific order of pattern presentation of the data in the training set. The same is true for the curves of Fig. 3(b) and (c). In these figures the × marks on one of the curves (test curve), correspond to the number of iterations at which training was stopped and the classification accuracy of Fuzzy ARTMAP on the validation and test sets was calculated. These performance results were used to generate the percentage of correct classification on the test and validation sets at various instances of the training process. One obvious observation that can be extracted from these figures is that the maxima and minima of the performance of Fuzzy ARTMAP on the validation set are coinciding with the maxima and minima of the performance of Fuzzy ARTMAP on the test set. This is an indication that the validation and test data sets are representative of the distribution that the data obey.

If we observe the results depicted in Tables 1– 3, we can draw some useful observations regarding the performance of Fuzzy ARTMAP under the three different modes of training.

1. The number of nodes created by Fuzzy ARTMAP trained under mode 3 (cross-validated training) is significantly smaller than the number of nodes created by Fuzzy ARTMAP trained under modes 1 (complete training) and 2 (one epoch of training). This observation is more pronounced for higher overlap datasets.
2. The generalization performance of Fuzzy ARTMAP trained under mode 3 (cross-validated training) is better than the generalization performance of Fuzzy ARTMAP trained under mode 1 (complete training) or mode 2 (one epoch of training).
3. The difference in the generalization performance between modes 3 (cross-validated training) and mode 2 (one epoch of training) is larger than the difference in the generalization performance between modes 3 and 1 (complete training).
4. The difference in the number of nodes created between mode 1 (complete training) and mode 3 (cross-validated training) is larger than the difference in the number of nodes created between mode 2 (one epoch of training) and mode 3.
5. The above observations are valid for all the dimensions (2, 5, 10) and all the number of distinct classes (2, 3) that we experimented with.

It is worth at this point to elaborate on observation 2 that states that mode 3 trained Fuzzy ARTMAP exhibits better generalization performance than mode 1 or mode 2 Fuzzy ARTMAP. The question that arises when a statement of this nature is made is whether the difference in performance between mode 3 and mode 1 or 2 Fuzzy ARTMAP is statistically significant. To answer this question let us focus on the test procedure that estimates trained Fuzzy ARTMAP accuracy. Each one of the datapoints of the test set (a total of *NS* datapoints) is presented to the trained Fuzzy ARTMAP and it produces a response from the trained Fuzzy ARTMAP network regarding its class label. If the response of Fuzzy ARTMAP is incorrect we say that we have committed a misclassification error. If we denote by *MNS* the number of test datapoints whose class label prediction by Fuzzy ARTMAP is incorrect then the ratio *MNS/NS* corresponds to the misclassification rate, and obviously

$$100\times\frac{MNS}{NS}=100-PCC \tag{11}$$

The number *MNS* is a random variable, whose distribution is the binomial distribution, with parameters *NS* and *p*, where *p* stands for the true misclassification rate of the trained Fuzzy ARTMAP (that is obviously unknown to us). The reason that the binomial distribution model is valid is because we are performing *NS* independent experiments (by presenting to the trained Fuzzy ARTMAP the *NS* randomly chosen test datapoints) and we are tabulating Fuzzy ARTMAP responses regarding the class label of each test datapoint presented to it. The response of Fuzzy ARTMAP is 1 if a misclassification of a test datapoint is observed and zero otherwise. The number of incorrect responses by Fuzzy ARTMAP is represented by *MNS* and, due to the above observations, *MNS* obeys the binomial distribution with parameters *NS* and *p*. It is a well known fact in probability that we can approximate the probability distribution of *MNS* with a Gaussian distribution of an appropriate mean and variance (provided that *NS* is a large enough number). Assuming that *NS* is large enough, let us denote by $MCNS(=MNS/NS)$ the misclassification rate exhibited by Fuzzy ARTMAP, where the misclassification rate is based on evaluating Fuzzy ARTMAP's performance on *NS* test datapoints. The random variable *MCNS* has mean *p* and variance $p(1-p)/NS$, and is approximately Gaussian for *NS* large. A probability of interest to us is:

$$\Pr[|MC_{NS}-p|<\epsilon] \tag{12}$$

The aforementioned probability represents the probability that the estimated misclassification Fuzzy ARTMAP rate *MCNS* is within ϵ from the true Fuzzy ARTMAP misclassification rate *p*. In practice, we want for a small enough ϵ to find how many test datapoints *NS* we need to present to Fuzzy ARTMAP in order to be α percent confident that *MCNS* and *p* are not more than ϵ apart from each other. For the purposes of this paper, we chose $\alpha=95$. By using the Gaussian approximation for the distribution of *MCNS* we can deduce, without a lot of difficulty, that

$$\Pr[|MC_{NS}-p|<\epsilon]=1-2Q\left(\frac{\epsilon\sqrt{NS}}{\sqrt{p(1-p)}}\right) \tag{13}$$

where $Q(x)$ is the well-known Gaussian error function that has been extensively tabulated. The above probability cannot be computed because *p* is unknown. However, it can be easily shown that $p(1-p)\leq 1/4$ for *p* in the unit interval. It then follows that for such *p*, $\sqrt{p(1-p)}\leq 1/2$, and since $Q(x)$ decreases with increasing argument

$$\Pr[|MC_{NS}-p|<\epsilon|]<1-2Q(2\epsilon\sqrt{NS}) \tag{14}$$

We want the left hand side of the above inequality to be equal to 0.95. It suffices then to choose *NS*, such that, $Q(2\epsilon\sqrt{NS})=(1-0.95)/2=0.025$. From appropriate tables, where $Q(x)$ has been tabulated, we can obtain that the argument of $Q(x)$ should be approximately 1.95, thus

$$2\epsilon\sqrt{NS}=1.95 \tag{15}$$

Solving for *NS* we obtain

$$NS=(0.98)^2/\epsilon^2=9506 \tag{16}$$

Table 4
Comparison of percentage of correct classification (*PCC*s) and number of nodes created for the three different Fuzzy ARTMAP training modes (1, 2, 3) using the Nursery database

Order	PCC_{v}	PCC_{tes}	$PCC^{\mathrm{c}}_{\mathrm{tes}}$	$PCC^{\mathrm{1EP}}_{\mathrm{tes}}$	PCC_{tr}	$PCC^{\mathrm{c}}_{\mathrm{tr}}$	$PCC^{\mathrm{1EP}}_{\mathrm{tr}}$	N_a	N^{c}_a	N^{1EP}_a
1	94.56	93.92	93.92	90.68	100	100	93.12	162	453	177
2	94.26	93.71	93.74	85.71	99.91	100	89.68	506	537	192
3	94.16	93.74	93.74	87.53	100	100	90.65	525	537	192
4	94.66	94.11	94.14	89.73	99.89	100	90.94	458	462	173
5	95.62	94.54	94.6	89.14	99.66	100	92.16	457	469	186
Avg	94.65	94.00	94.03	88.56	99.89	100	91.31	421.36	486.2	181

The last equality in Eq. (16) was produced by setting $\epsilon = 0.01$. This justifies our choice of testing the trained Fuzzy ARTMAP architectures with 5000 datapoints per class. As a reminder, we examined trained Fuzzy ARTMAP with two, or three classes, hence in the worst possible case we are testing Fuzzy ARTMAP with $NS = 10{,}000$ test datapoints. Consequently, the true PCC_{tes} results are, with confidence 95%, within ±1% from the values reported in the tables. Knowing also that our aforementioned claims are based on an upper bound for the quantity $p(1-p)$, occurring at $p = 1/2$, it gives us even bigger confidence in believing that the performance difference between Fuzzy ARTMAP trained under mode 3 and Fuzzy ARTMAP trained under modes 1 and 2 is statistically significant. Specifically, let us refer to the $PCC_{\mathrm{tes}} = 73.59\%$ and $PCC^{\mathrm{c}}_{\mathrm{tes}} = 65.8\%$ results of row 1 of Table 3(c). The aforementioned analysis tells us that with 95% confidence PCC_{tes} lies in the interval [72.59%, 74.59%], while $PCC^{\mathrm{c}}_{\mathrm{tes}}$ lies in the interval [64.8%, 66.8%]. Hence, PCC_{tes} is statistically significantly higher than $PCC^{\mathrm{c}}_{\mathrm{tes}}$, at least for the most of the higher overlap cases. For the medium overlap cases we see differences of PCC_{tes} and $PCC^{\mathrm{c}}_{\mathrm{tes}}$ of the order of 2%. Hence, the statistical significance of these differences is less. We could have increased the number of datapoints per class from 5000 to higher values to increase the statistical significance of the differences in *PCC*s for the three modes of training. We decided against doing that because it would have led to exceedingly slow Fuzzy ARTMAP training phases. In a similar fashion, the statistical significance of the PCC_{tes}s and $PCC^{\mathrm{c}}_{\mathrm{tes}}$s for the low overlap cases is even less than that for the medium overlap cases. Nevertheless, we believe that for the low overlap cases the PCC_{tes}s and $PCC^{\mathrm{c}}_{\mathrm{tes}}$s are very close to each other. The major differences for the low overlap cases are observed in the number of nodes created (see Table 1).

Due to the observations 1–5, made earlier, we can make the claim that overtraining in Fuzzy ARTMAP does occur, and in these cases cross-validation is a legitimate procedure that allows us to create smaller Fuzzy ARTMAP networks with better generalization performance. It is worth mentioning, though, that there is a price to pay when cross-validation is employed in Fuzzy ARTMAP training. The price is increased computational complexity during training. This price, though, might be worth paying to avoid the creation of oversized Fuzzy ARTMAP networks (e.g. see the comparison of N_a, N^{c}_a, N^{1EP}_a in Table 3).

4.2. Real databases

As we have mentioned earlier, there is an advantage of training/validating and testing Fuzzy ARTMAP with artificial databases. The reason is that we can easily change the number of distinct classes, the dimensionality of the input patterns and the amount of overlap of patterns belonging to different classes. It is worth, though, examining the comparative performance of Fuzzy ARTMAP and Fuzzy ARTMAP with cross-validation for some real databases as well. We chose the real databases from the well-known UCI Repository (Murphy and Aha, 1994). From the available collection of databases there, we chose databases that had a relatively large number of datapoints to satisfy the assumption in this paper that we are dealing with large databases. The databases chosen to experiment with were the Nursery database and the Letters database.

4.2.1. Nursery database

This database has five distinct classes with a total of 12,960 datapoints. The dimensionality of the input patterns is 8. We split the data into a training set (6480 points), validation set (3239 points) and test set (3241 points). For this collection of training, validation and test sets we trained Fuzzy ARTMAP for five different orders of training pattern presentations. Fuzzy ARTMAP was trained and tested with the Nursery data for all the modes of training discussed in this paper (i.e. modes 1, 2 and 3). One of the differences between the mode 3 training here and the one reported for the artificial databases is that cross-validation here is performed for all epochs of training. The results are reported in Table 4. The entries in Table 4 correspond to: PCC_{v}, PCC_{tes}, $PCC^{\mathrm{c}}_{\mathrm{tes}}$, $PCC^{\mathrm{1EP}}_{\mathrm{tes}}$, PCC_{tr}, $PCC^{\mathrm{c}}_{\mathrm{tr}}$, $PCC^{\mathrm{1EP}}_{\mathrm{tr}}$, N_a, N^{c}_a, p N^{1EP}_a for each one of the five orders of training pattern presentations in the training set. These entries have the same interpretation as the entries of Table 3. In Fig. 4 we also show the PCC_{v} and PCC_{tes} for one of these five orders of pattern presentation as training in Fuzzy ARTMAP progresses. The figure validates our confidence that the distributions of the data in the validation and test sets are similar.

An obvious observation from the results reported in Table 4 is that the maximum validation performance of Fuzzy ARTMAP happens close to the completion of training (compare PCC_{tr} and $PCC_{\mathrm{tr}}^{\mathrm{c}}$ in Table 4). There is where we also observe a test set performance that is very close to the maximum test set performance (observed at the completion of training). This is an indication that there is no significant overtraining when Fuzzy ARTMAP is trained with the Nursery data, and it might be worth training Fuzzy ARTMAP to completion in this case. Of course, an increased generalization average performance of 5% (at the completion of training) results in a Fuzzy ARTMAP architecture that has 2.7 times as many nodes as a Fuzzy ARTMAP trained for only one epoch (see Table 4).

4.2.2. Letters database

This database has 26 distinct classes (letters A–Z) with a total of 20,000 datapoints. The dimensionality of the input patterns is 16. We split the data into a training set (10,009 points), validation set (5009 points) and test set (4984 points). For this collection of training, validation and test sets we trained Fuzzy ARTMAP for five different orders of training pattern presentations. Fuzzy ARTMAP was trained and tested with the Letters data for all the modes of training discussed in this paper (i.e. modes 1, 2 and 3). One of the differences between the mode 3 training here and the one reported for the artificial databases is that cross-validation here is performed for all epochs of training. The results are reported in Table 5. The entries in Table 5 correspond to: PCC_{v}, PCC_{tes}, $PCC_{\mathrm{tes}}^{\mathrm{c}}$, $PCC_{\mathrm{tes}}^{\mathrm{1EP}}$, PCC_{tr}, $PCC_{\mathrm{tr}}^{\mathrm{c}}$, $PCC_{\mathrm{tr}}^{\mathrm{1EP}}$, N_a, $N^{\mathrm{c}}{}_a$, N_a^{1EP} for each one of the five orders of training pattern presentations in the training set. These entries have the same interpretation as the entries of Tables 3 and 4. In Fig. 5 we also show the PCC_{v} and PCC_{tes} for one of these five orders of pattern presentation as training in Fuzzy ARTMAP progresses. The figure validates our confidence that the distributions of the data in the validation and test sets are similar.

An obvious observation from the results reported in Table 5 is that the maximum validation performance of Fuzzy ARTMAP happens at the completion of training (compare PCC_{tr} and $PCC_{\mathrm{tr}}^{\mathrm{c}}$ in Table 5). There is where we also observe the maximum test set performance. This is an indication that there is no overtraining when Fuzzy ARTMAP is trained with the Letters database. It is also worth noticing here that the average generalization performance at the completion of training is 2.7% higher than the average generalization performance achieved after one epoch of training. This increased generalization performance comes at virtually no expense on the number of nodes created by Fuzzy ARTMAP by the completion of training compared to training for one epoch. Consequently, this is an example of a database that is worth training until we achieve a 100% performance on the training set.

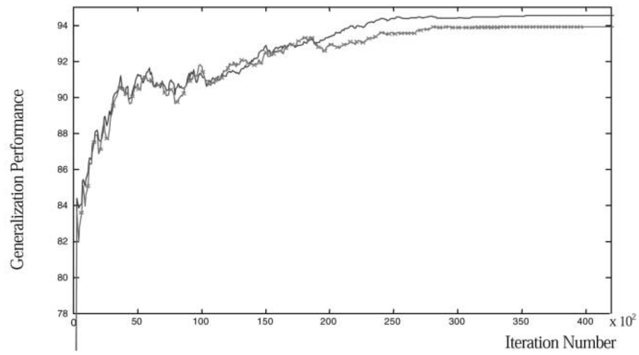

Fig. 4. Percentage of correct classification on the validation set (–) and testing set (×) of a trained Fuzzy ARTMAP network with data of different degrees of overlap (Nursery database).

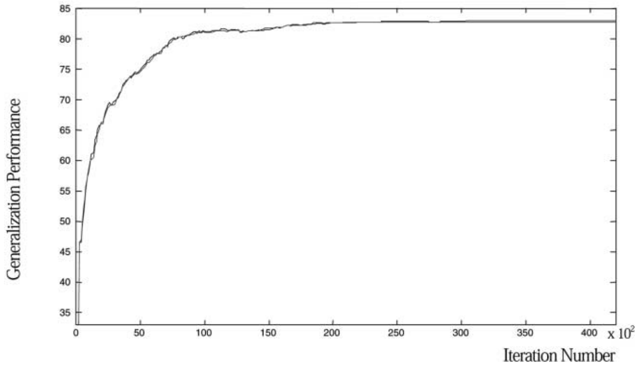

Fig. 5. Percentage of correct classification on the validation set (–) and testing set (×) of a trained Fuzzy ARTMAP network with data of different degrees of overlap (Letters database).

5. Conclusions

In this paper, we investigated the relative performance of

Table 5
Comparison of percentage of correct classification (*PCC*s) and number of nodes created for the three different Fuzzy ARTMAP training modes (1, 2, 3) using the Letters database

Order	PCC_{v}	PCC_{tes}	$PCC_{\mathrm{tes}}^{\mathrm{c}}$	$PCC_{\mathrm{tes}}^{\mathrm{1EP}}$	PCC_{tr}	$PCC_{\mathrm{tr}}^{\mathrm{c}}$	$PCC_{\mathrm{tr}}^{\mathrm{1EP}}$	N_a	N_a^{c}	N_a^{1EP}
1	82.99	82.83	82.83	81.01	100	100	96.05	710	710	659
2	83.13	82.77	82.77	80.26	100	100	95.3	678	678	619
3	83.31	83.27	83.49	81.49	99.35	100	96.11	667	684	632
4	83.51	83.41	83.43	80.99	99.99	100	95.96	691	695	638
5	84.11	83.03	83.03	80.32	100	100	95.5	712	712	651
Avg	83.41	83.06	83.11	80.81	99.87	100	95.70	691.6	695.8	639.8

Fuzzy ARTMAP trained to completion, or trained for one epoch, compared to the performance of Fuzzy ARTMAP trained until the maximum performance on a validation set is achieved. The results on the artificial databases, where we could control the amount of data used, the dimensionality of the input patterns and the degree of overlap of data belonging to different classes, indicate that cross-validation help us discover a Fuzzy ARTMAP network with increased generalization and significantly reduced number of nodes. These conclusions were more pronounced as we moved from databases of low overlap to databases of higher overlap. The results with the real databases are also encouraging. Cross-validation allows us to determine, in a straightforward fashion, whether overtraining of Fuzzy ARTMAP occurs or not during the training process. Overtraining of a neural network architecture manifests itself in two different ways for Fuzzy ARTMAP: decreased generalization performance as training progresses and/or creation of larger size Fuzzy ARTMAP architectures as training progresses. For example, with the Nursery database we concluded that the Fuzzy ARTMAP generalization performance keeps improving until the completion of training, at the expense of creating a larger Fuzzy ARTMAP architecture. On the other hand, with the Letters database we observed that the Fuzzy ARTMAP generalization performance keeps improving until the completion of training without creating a larger size architecture. In both of these cases, by using cross-validation we were able to make an educated guess of when to stop the training process.

We performed experiments additional to the ones carefully described in Section 4. For example, we experimented with some of the artificial databases and the real databases for choice parameter β_a value equal to 1; $\beta_a = 1$ has been extensively used in simulations of Fuzzy ARTMAP reported in the literature. The results obtained from these experiments with the artificial databases are of a nature similar to the ones reported in observations 1–5 of Section 4.1. The only difference is that for $\beta_a = 1$ the mode 1 and mode 2 Fuzzy ARTMAP create a lot more nodes than their counterparts for $\beta_a = 0.01$. As a consequence, the compression ratios attained by the mode 3 Fuzzy ARTMAP compared to the modes 1 and 2 Fuzzy ARTMAP, when $\beta_a = 1$, are higher than the ones reported in Tables 1–3, where $\beta_a = 0.01$. The results obtained from the experiments with the real databases, when $\beta_a = 1$, are also of similar nature to the ones obtained when $\beta_a = 0.01$ (e.g. no overtraining is observed). Furthermore, we experimented with a mode 3 Fuzzy ARTMAP using a validation period of 10 instead of 100 that was used in all the experiments reported in Section 4. As a reminder, the mode 3 Fuzzy ARTMAP training is stopped at specific iteration instances and its performance on a validation set is checked. The difference between two such consecutive iteration instances is referred to as the validation period. For the real databases results, we did not observe any differences by using a smaller validation period. For the artificial databases results, we noticed that mode 3 Fuzzy ARTMAP attained a better generalization performance PCC_{tes} when the validation period was 10 instead of 100. Obviously, the computational overhead imposed when the validation period is 10 is 10 times higher than the one imposed when the validation period is 100.

References

Amari, S., Murata, N., Muller, K., Finke, M., & Yang, H. (1996). Statistical theory of overtraining—is cross-validation asymptotically effective? *Advances in Neural Information Processing Systems*, *8*, 176–182.

Amari, S., Murata, N., Muller, K., Finke, M., & Yang, H. (1997). Asymptotic statistical theory of overtraining and cross-validation. *IEEE Transactions on Neural Networks*, *8* (5), 985–996.

Anders, U., & Korn, O. (1999). Model selection in neural networks. *Neural Networks*, *12* (2), 309–323.

Carpenter, G. A., & Markuzon, N. (1998). ARTMAP-IC and medical diagnosis: instance counting and inconsistent cases. *Neural Networks*, *11* (2), 323–336.

Carpenter, G. A., & Ross, W. D. (1995). ART-EMAP: a neural network architecture for object recognition by evidence accumulation. *IEEE Transactions on Neural Networks*, *6*, 805–818.

Carpenter, G. A., Grossberg, S., Markuzon, N., Reynolds, J. H., & Rosen, D. B. (1992a). Fuzzy ARTMAP: a neural network architecture for incremental supervised learning of analog multi-dimensional maps. *IEEE Transactions on Neural Networks*, *3* (5), 698–713.

Carpenter, G. A., Grossberg, S., & Reynolds, J. H. (1991). ARTMAP: supervised real-time learning and classification of non-stationary data by a self-organizing neural network. *Neural Networks*, *4* (5), 565–588.

Carpenter, G. A., Grossberg, S., & Rosen, D. B. (1992b). Fuzzy ART: fast stable learning and categorization of analog patterns by adaptive resonance systems. *Neural Networks*, *4*, 759–771.

Charalampidis, D., Georgiopoulos, M., & Kasparis, T. (2000). Classification of noisy signals using Fuzzy ARTMAP neural networks. In *Proceedings of the International Joint Conference in Neural Networks (IJCNN) 2000, Como, Italy, July 24–28, 2000* (pp. VI53–VI59).

Dietrich, T. G. (1998). Approximate statistical test for comparing supervised classification learning algorithms. *Neural Computation*, *10*, 1895–1923.

Georgiopoulos, M., Dagher, I., Bebis, G., & Heileman, G. L. (1999). Properties of learning of a Fuzzy ART variant. *Neural Networks*, *12* (6), 837–850.

Georgiopoulos, M., Fernlund, H., Bebis, G., & Heileman, G. L. (1996). Order of search in Fuzzy ART and Fuzzy ARTMAP. *Neural Networks*, *9* (9), 1541–1559.

Georgiopoulos, M., Heileman, G. L., & Huang, J. (1991). Properties of learning related to pattern diversity in ART1. *Neural Networks*, *4* (6), 751–758.

Georgiopoulos, M., Huang, J., & Heileman, G. L. (1994). Properties of learning in ARTMAP. *Neural Networks*, *7* (3), 495–506.

Gomez Sanchez E., Dimitriadis Y. A., Cano Izquierdo J. M., & Lopez Colorado J. (2000). MicroARTMAP: use of mutual information for category reduction in Fuzzy ARTMAP. In *Proceedings of the International Joint Conference in Neural Networks (IJCNN) 2000, Como, Italy, July 24–28, 2000* (pp. VI47–VI52).

Healy, M. J., Caudell, T. P., & Smith, S. D. G. (1993). A neural network architecture for patterns sequence verification through inferencing. *IEEE Transactions on Neural Networks*, *4* (1), 9–20.

Huang, J., Georgiopoulos, M., & Heileman, G. L. (1995). *Fuzzy art properties. Neural Networks*, *8* (2), 203–213.

Kohavi, R. (1995). A study of cross-validation and bootstrap for accuracy estimation and model selection. In *Proceedings of the 14th International Joint Conference on Artificial Intelligence* (pp. 1137–1143).

Moore, B. (1988). ART1 and pattern clustering. In D. Touretzky & G. H.

Sejnowski (Eds.), *Proceedings of the 1988 Connectionist Summer School, San Mateo, CA* (pp. 174–185). Morgan Kaufmann.

Murphy, P., & Aha, D. (1994). *UCI Repository of Machine Learning Databases*, Technical Report, http://www.ics.edu/mlearn/MLRepository.html, Department of Computer Science, University of California Irvine.

Prechelt, L. (1998). Automatic early stopping using cross-validation: quantifying the criteria. *Neural Networks*, *11* (4), 761–767.

Rumelhart, D. E., Hinton, G. E., & Williams, R. J. (1986). PDP book. In D. E. Rumelhart & J. J. McClelland, *Learning internal representations by error propagation*, vol. 1. Cambridge, MA: MIT Press.

Simpson, P. K. (1992). Fuzzy Min-Max neural networks: Part I Classification. *IEEE Transactions on Fuzzy Systems*, *3* (5), 776–786.

Simpson, P. K. (1993). Fuzzy Min-Max neural networks: Part II. Clustering. *IEEE Transactions on Fuzzy Systems*, *1* (1), 32–45.

Stone, M. (1974). Cross-validatory choice and assessment of statistical predictions. *Journal of the Royal Statistical Society Series (Methodological)*, *36* (2), 111–147.

Stone, M. (1977). Asymptotics for and against cross-validation, *Biometrica 64* (1), 29–35.

Vertzi, S., Heileman, G. L., Georgiopoulos, M., & Healy, M. J. (1998). Boosting the performance of ARTMAP. In *Proceedings of the 1998 International Joint Conference on Neural Networks (IJCNN-98), Anchorage, AL, June 1998* (pp. 396–401).

Williamson, J. R. (1996). Gaussian ARTMAP a neural network for fast incremental learning of noisy multi-dimensional maps. *Neural Networks*, *9* (5), 881–897.