A Fuzzy ARTMAP Nonparametric Probability Estimator for Nonstationary Pattern Recognition Problems

Gail A. Carpenter, *Member, IEEE,* Stephen Grossberg, and John H. Reynolds, *Member, IEEE*

Abstract—**An incremental, nonparametric probability estimation procedure using the fuzzy ARTMAP (adaptive resonance theory-supervised predictive mapping) neural network is introduced. In the slow-learning mode, fuzzy ARTMAP searches for patterns of data on which to build ever more accurate estimates. In max-nodes mode, the network initially learns a fixed number of categories, and weights are then adjusted gradually.**

I. Fuzzy ARTMAP for Probability Estimation

MANY pattern recognition applications require an estimate of the probability that an input belongs to a given class. In a medical database, for example, a set of measurements can be used to estimate the probability that a patient will require a long stay in the hospital. Different groups of diagnostic factors may be associated with a single outcome, and it is possible that no single combination of variables forms a unique set of predictors. Fuzzy ARTMAP (adaptive resonance theory-supervised predictive mapping), [1], [2] discussed in Section II, is a neural network that automatically selects complex combinations of factors on which to build accurate predictions for application to problems such as medical prediction and handwritten character recognition [3]–[5]. Fuzzy ARTMAP is able to create a stable memory structure even with fast, on-line learning. With fast learning, the network would regard each on-line training point as potentially informative, possibly an important rare case, and record its prediction in the set of learned categories. In this training mode, however, noisy data can lead to category proliferation.

A procedure that uses fuzzy ARTMAP slow learning for probability estimation in a noisy input environment is developed here. Unlike parametric probability estimators, fuzzy ARTMAP does not depend on *a priori* assumptions about the underlying data. The network can make accurate probability estimates even when the underlying distributions are unknown and when data sets arrive incrementally. Fuzzy ARTMAP on-line computations achieve both accurate probability estimates and good code compression by partitioning the input space into categories. Large or small recognition categories form to generate the best output predictions, and a variable number of recognition categories may predict each output. Categories evolve through a hypothesis testing process that incrementally incorporates information about each pattern into a knowledge base. If the system encounters a region of input space that includes several small clusters of inputs from different classes, it breaks those regions into subregions and makes a probability estimate for each subregion. ARTMAP can thus make broad, efficient generalizations, but also reduces false alarms by identifying rare or exceptional cases. Methods that try to fit the data to an assumed but incorrect distribution can fail to identify these exceptions.

Simulations demonstrate that the fuzzy ARTMAP probability estimation procedure is robust, performing well in problems with different types of input distributions. Two variants of this method are described, the slow-learning mode in Section III and the max-nodes mode in Section IV. In slow-learning mode, the system grows incrementally until it achieves a good fit to the underlying probability density function. In max-nodes mode, the user specifies an upper bound on network size. After it has reached this size the network stops growing, but additional training data can still be incorporated into the existing network to improve its probability estimates. Simulations of three probability estimation problems compare performance of both modes of fuzzy ARTMAP to that of Bayesian estimation. The three tasks requiring probability estimation for a simple two-Gaussian distribution are discussed in Section V, a trimodal distribution is discussed in Section VI, and a difficult problem in which inputs to each class form distributions that are 97-modal, with modes falling on two intertwined spirals, is discussed in Section VII. Fuzzy ARTMAP provides accurate estimates in both modes for all three tasks. Finally, Section VIII includes a discussion of the ARTMAP algorithmic variations, and the Appendix illustrates slow learning with a computational example.

II. Fuzzy ARTMAP

Fuzzy ARTMAP (Fig. 1) includes a pair of ART modules [6] (ART_a and ART_b) that create stable recognition categories

Manuscript received August 6, 1993; revised November 12, 1994 and March 21, 1995. This work was supported in part by the Air Force Office of Scientific Research (AFOSR F49620-92-J-0175 and AFOSR 90-0225), the Advanced Research Projects Agency (ONR N00014-92-J-4015), the National Science Foundation (NSF IRI 94-00530 and NSF IRI-90-01659), and the Office of Naval Research (ONR N00014-91-J-4100, ONR N00014–95–0409, and ONR N00014–95–0657).

The authors are with the Center for Adaptive Systems and Department of Cognitive and Neural Systems, Boston University, Boston, MA 02215 USA.

IEEE Log Number 9413442.

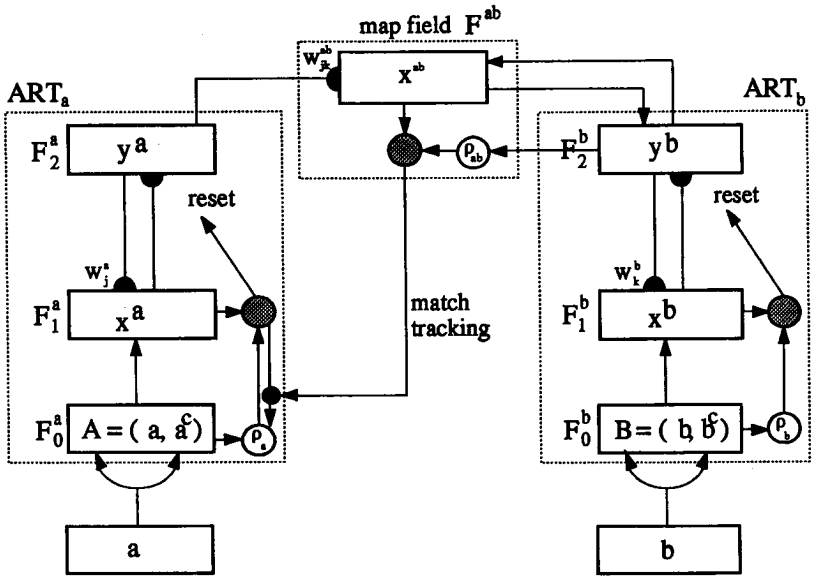

Fig. 1. Fuzzy ARTMAP architecture [1]. The ART_a complement coding preprocessor transforms the M_a-vector $\boldsymbol{a}$ into the $2M_a$-vector $\boldsymbol{A} = (\boldsymbol{a}, \boldsymbol{a}^c)$ at the ART_a field F_0^a, where $\boldsymbol{a}^c = \boldsymbol{1} - \boldsymbol{a}$. Vector $\boldsymbol{A}$ is the input to the ART_a field F_1^a. Similarly, the input to F_1^b is the $2M_b$-vector $\boldsymbol{B} = (\boldsymbol{b}, \boldsymbol{b}^c)$. When a prediction by ART_a is disconfirmed at ART_b, inhibition of the map field F^{ab} activates a match tracking process. Match tracking raises the ART_a vigilance (ρ_a) to just above the F_1^a-to-F_0^a match ratio $|\boldsymbol{x}^a|/|\boldsymbol{A}|$. This triggers an ART_a search that leads to activation of either an ART_a category that correctly predicts $\boldsymbol{b}$ or to a previously uncommitted ART_a category node.

in response to arbitrary sequences of input patterns. During supervised learning, ART_a receives a stream $\{\boldsymbol{a}^{(p)}\}$ of input patterns and ART_b also receives a stream $\{\boldsymbol{b}^{(p)}\}$ of patterns, where $\boldsymbol{b}^{(p)}$ is the correct prediction given $\boldsymbol{a}^{(p)}$. These modules are linked by an associative learning network and an internal controller that ensures autonomous system operation in real time. The controller is designed to create the minimum number of ART_a recognition categories, or "hidden units," needed to meet accuracy criteria.

Parameter ρ_a calibrates the minimum confidence that ART_a must have in a recognition category, or hypothesis, activated by an input $\boldsymbol{a}^{(p)}$ for ART_a to accept that category, rather than search for a better one through the automatically controlled process of hypothesis testing. Lower values of ρ_a enable larger categories to form. These lower ρ_a values lead to broader generalization and a higher degree of code compression. A predictive failure at ART_b increases ρ_a by the minimum amount needed to trigger hypothesis testing at ART_a, using a mechanism called match tracking. Match tracking sacrifices just enough generalization to correct a predictive error. Hypothesis testing leads to the selection of a new ART_a category, which focuses attention on a new cluster of $\boldsymbol{a}^{(p)}$ input features that is better able to predict $\boldsymbol{b}^{(p)}$. Match tracking allows a single ARTMAP system to learn a different prediction for a rare event than for a cloud of similar frequent events in which it is embedded. The fuzzy ARTMAP algorithm [1] scales to arbitrary dimensions. Low-dimensional simulation examples illustrate the algorithmic variations introduced here.

III. Slow-Learning Mode

In the slow-learning mode, fuzzy ARTMAP slowly updates its map field weights to estimate the probability that an input belongs to a given output class. In particular, when an input activates an ART_a category at level F_2^a, the size of the weight in the pathway from the F_2^a category to a map field category node (Fig. 1) provides an estimate of the probability that the input belongs to the output class coded by the map field node. During supervised learning, the strength of the weight projecting from the selected ART_a category to the correct ART_b category is increased, while the strengths of the weights to other ART_b categories are decreased. A map field vigilance parameter (ρ_{ab}) calibrates the degree of novelty, or 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 map field to the active ART_b category is smaller than ρ_{ab}, the system responds to the unexpected outcome through match tracking, which triggers an ART_a search for a new F_2^a recognition category.

Once an ART_a category (J) is chosen whose prediction of the correct ART_b category is strong enough, match tracking is disengaged, and the network is said to be in a state of resonance. Then, learning proceeds at ART_a according to the fuzzy ART [7] fast learning equation. Map field learning obeys the equation

$$(w_{jk}^{ab})^{\text{new}} = \begin{cases} (1-\beta_{ab})(w_{jk}^{ab})^{\text{old}} + \beta_{ab} x_k^{ab} & \text{if } j = J \\ (w_{jk}^{ab})^{\text{old}} & \text{if } j \neq J \end{cases} \quad (1)$$

where w_{jk}^{ab} is the map field weight projecting to map field node k from the ART_a node j, with $w_{jk}^{ab}(0) = 1$ and where map field activity $x_k^{ab} = 1$ when k is the correct ART_b category and $x_k^{ab} = 0$ otherwise. The map field learning parameter β_{ab} determines the rate of change of the map field weights. Small values of β_{ab} cause the system to base its probability estimate

on a long-term average of its experience, while values of β_{ab} near one allow adaptation to a rapidly changing environment. The Appendix includes a computational example of this slow-learning process.

IV. Max-Nodes Mode

For large-scale applications, it may be necessary to limit the size of the network for computational efficiency. In such circumstances, ARTMAP can operate in a max-nodes mode, in which the user specifies a maximum number of F_2^a category nodes. This method sets map field vigilance ρ_{ab} to one during early training, to establish an initial categorical mapping. After the maximum number of ART_a categories has been reached, ρ_{ab} is set to zero, so match tracking never occurs in response to a predictive mismatch. With ρ_{ab} initially equal to one, match tracking will be triggered whenever a predictive error occurs. This initial "critical period" establishes a tessellation of the input space associating each region with one output class. After ρ_{ab} is lowered to zero, map field weights slowly adjust their estimates of the probability that a member of a given ART_a category belongs to a given ART_b class. With $\rho_{ab} = 0$ and $\beta_{ab} < 1$, the rapid partition established with $\rho_{ab} = 1$ is fine tuned via slow learning.

V. Simulation: Two Gaussians

In a two-Gaussian probability estimation task, inputs from two classes are drawn from two overlapping distributions (Fig. 2). For this task, a simple, two-Gaussian model makes accurate probability estimates, with the task reduced to estimating the parameters of the underlying distributions.

In the probability estimation task depicted in Fig. 2, the input points in a unit square were drawn from two Gaussian distributions centered at $\mu_1 = (0.5, 0.75)$ and $\mu_2 = (0.5, 0.25)$, with covariances $\sigma_1^2 = 0.15, \sigma_{12} = 0, \sigma_{21} = 0$, and $\sigma_2^2 = 0.15$. Fig. 2(a) indicates the size of the two Gaussians, with circles centered at μ_1 (white) and μ_2 (black), with radii $2\sigma_i^2 = 0.3$. Approximately 95% of the class 1 patterns fall within the white circle and 95% of the class 2 patterns within the black circle. Fig. 2(b) shows the actual training data, drawn from the two distributions with equal probability. The 520 white points belong to class 1 and the 480 black points belong to class 2. Fig. 2(c) shows the actual probability that a pattern falling at each point in the unit square will belong to each of the two classes. Patterns falling in lighter regions are more likely to belong to class 1, while those in darker regions are more likely to belong to class 2. These probabilities represent the ideal estimate calculated using Bayes' rule

$$P(c|\boldsymbol{a}) = \frac{p(\boldsymbol{a}|c)P(c)}{p(\boldsymbol{a})} \tag{2}$$

where

$$p(\boldsymbol{a}) = \sum_{c=1}^{2} p(\boldsymbol{a}|c)P(c). \tag{3}$$

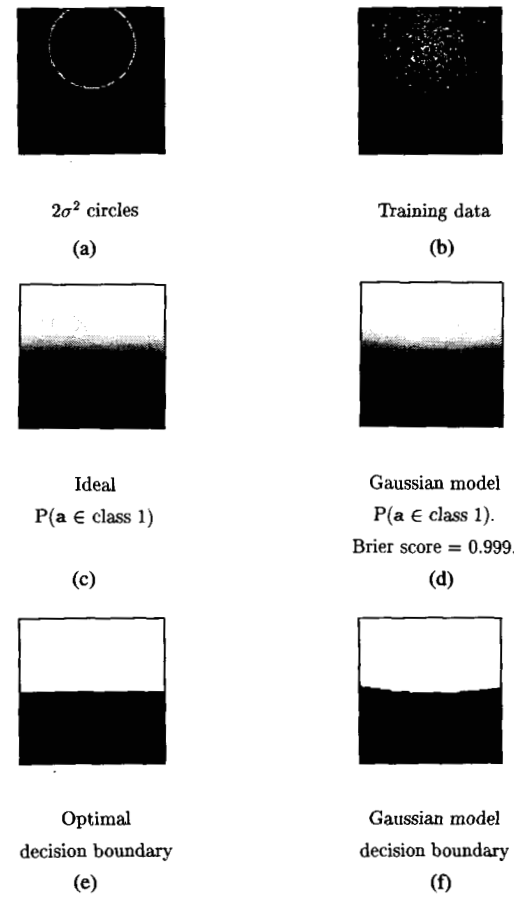

Fig. 2. Two overlapping Gaussians with equal *a priori* probabilities. (a) Circles around Gaussian means, each with radius $2\sigma^2$. (b) Actual training data (1000 points). (c) Actual conditional probability $P(\boldsymbol{a} \in \text{class 1})$. Points falling in a lighter region are more likely to belong to class 1, darker to class 2. (d) Gaussian model estimated conditional probabilities. (e) Optimal decision boundary. Points appearing in white area are assigned to class 1, black to class 2. (f) Gaussian model decision boundary.

In (2) and (3), $c = 1, 2$ is the class index, $P(c|\boldsymbol{a})$ is the *a posteriori* probability that pattern $\boldsymbol{a}$ belongs to class c, $p(\boldsymbol{a}|c)$ is the probability density function of $\boldsymbol{a}$ given that the class is c, and $P(c)$ is the *a priori* probability of class c. Fig. 2(d) shows the probability estimate computed by assuming that the two distributions are Gaussian and estimating their means and covariance matrices from the training data. Since the input set exactly meets the distributional assumptions of the two-Gaussian model, model estimates are very close to the ideal solution [Fig. 2(c)]. Fig. 2(e) shows the decision regions of the ideal maximum *a posteriori* classifier derived from the probability estimate of Fig. 2(c). Points in the white region are more likely to belong to class $c = 1$, and points in the black region are assigned to class $c = 2$. These classification regions will minimize the expected number of misclassifications. Fig. 2(f) shows the corresponding decision regions of the two-Gaussian model. As expected, the decision region shown in Fig. 2(f) appears to be very similar to the ideal solution shown in Fig. 2(e). The degree of similarity can be quantified as follows.

The performance of a two-class probability estimator can be quantified by calculating its average Brier score. The Brier score is a value which reflects how well a probability estimator

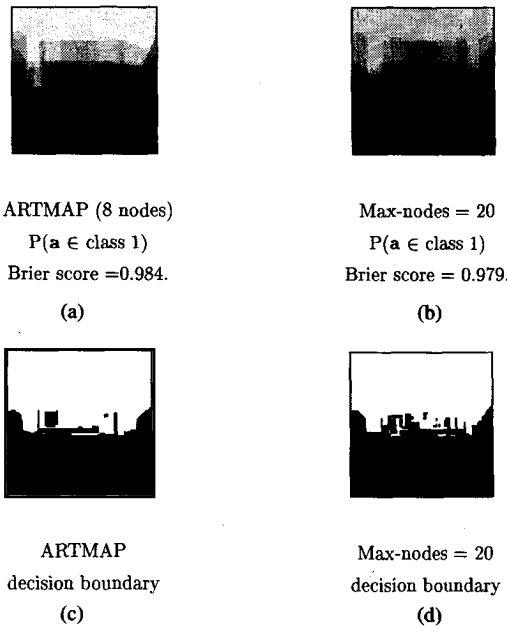

Fig. 3. Fuzzy ARTMAP model estimated conditional probabilities and decision boundary for the two-Gaussian problem. (a) Slow-learning model estimated conditional probability, averaged over nine orderings of the training data. Map field learning rate $\beta_{ab} = 0.02$, map field matching criterion $\rho_{ab} = 0.75$. (b) Max-nodes = 20 model estimated conditional probability, averaged over nine orderings of the training data: $\beta_{ab} = 0.01$ and $\rho_{ab} = 1$ until 20 ART_a nodes are committed, then $\rho_{ab} = 0$. (c) Slow-learning ARTMAP model decision boundary. (d) Max-nodes = 20 ARTMAP model decision boundary.

approximates the true probability of an output. The score $u(q,p)$ is a function of the estimated probability (q) and the true probability (p) according to the equation

$$u(q,p) = 1 - (q-p)^2. \tag{4}$$

This function is maximized at $u(q,p) = 1$, when the estimated probability is equal to the true probability, and minimized at $u(q,p) = 0$, when the estimated probability differs from the true probability by one. The average Brier score of the two-Gaussian model was calculated for 10 000 points evenly spaced on a grid covering the unit square. This average score was very high (0.999), indicating that the two-Gaussian model provides a good estimate of the distribution of the training data.

Fuzzy ARTMAP was also able to estimate probabilities for the two-Gaussian problem (Fig. 3), although not as efficiently as a system that is ideally suited to the task via *a priori* knowledge of probability distributions. Fig. 3(a) shows fuzzy ARTMAP probability estimates with slow learning, computed as the strength of the weight w_{Jc}^{ab} projecting from the winning ART_a node to ART_b node c $(c = 1, 2)$, divided by the sum of the weights projecting from the winning ART_a node $(w_{J1}^{ab} + w_{J2}^{ab})$ where

$$P(c|\mathbf{a}) = \frac{w_{Jc}^{ab}}{w_{J1}^{ab} + w_{J2}^{ab}}. \tag{5}$$

Performance was robust for a broad range of parameter choices. In simulations, the learning rate parameter β_{ab} in (1) was set to 0.02. On each pattern presentation, map field weights were then moved 2% of the way to zero or one, depending on which node was selected at ART_b. Map field vigilance ρ_{ab} was set to 0.75, so match tracking was engaged whenever the size of the weight projecting from the winning ART_a node J to the winning ART_b node c was less than 0.75. The results were averaged over nine independent orderings of the training data. Since ARTMAP is a fast incremental learning algorithm, the trained network weights vary with the order of the input presentation. By averaging estimates over several different orderings of a single data set, order dependence is reduced. On average, the system created eight nodes, for a data compression ratio of 125 : 1 and Brier score of 0.984. Fig. 3(b) shows the average probability estimate of an ARTMAP max-nodes system, with $\beta_{ab} = 0.01$. The map field vigilance, ρ_{ab}, equals one until 20 nodes are created, after which ρ_{ab} equals zero. This method achieves a data compression ratio of 50 : 1 and Brier score of 0.979. Twenty nodes were sufficient because the task was very simple. More complicated tasks tend to require a larger upper bound on the number of nodes. Even though it did not incorporate any knowledge of the underlying probability distribution, fuzzy ARTMAP achieves a good probability estimate.

VI. Simulation: Multimodal Distributions

A multimodal distribution problem with two output classes violates the *a priori* assumptions of the two-Gaussian model. For example, in the task shown in Fig. 4, inputs were drawn from each of six Gaussian distributions arranged in a ring [Fig. 4(a)]. White circles correspond to the Gaussians whose patterns belong to class 1, and black circles correspond to class 2. Fig. 4(b) shows the training inputs, which were drawn from the six distributions with equal probability. The 510 white points belong to class 1, and the 490 black points belong to class 2. Fig. 4(c) shows the actual probability that a pattern falling at each point in the unit square will belong to each of the two classes, calculated using Bayes' rule. As in Fig. 2(c), patterns falling in lighter regions are more likely to belong to class 1, while those in darker regions are more likely to belong to class 2. Fig. 4(d) shows the estimate of the simple two-Gaussian model, which assigns a probability of about 0.5 to each point. Fig. 4(e) and (f) compare the ideal classification with that of the two-Gaussian model.

Fuzzy ARTMAP (Fig. 5) identified the six regions corresponding to each of the six Gaussians in Fig. 4 and accurately estimated the class probabilities in each. The probability estimates of the system, averaged over nine orderings of the same data, are shown for the slow-learning mode [Fig. 5(a)] and the max-nodes mode [Fig. 5(b)]. The same ARTMAP system parameters were used as in Fig. 3. Estimated probabilities appear as three lighter and three darker areas, corresponding to classes 1 and 2. The *a posteriori* decision regions produced by each system [Fig. 5(c) and (d)] correctly identified large regions corresponding to the six ideal regions [Fig. 4(e)]. Differences were concentrated near the borders of the regions, where actual probabilities are near chance. On average, 26 ART_a nodes were created in slow-learning mode, for a data-

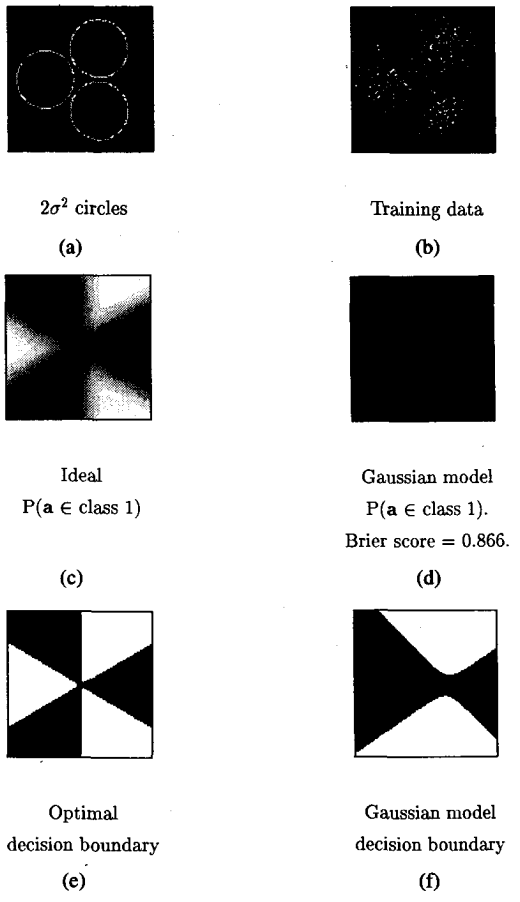

Fig. 4. Six overlapping Gaussians with equal *a priori* probabilities. (a) Circles around Gaussian means, with radius $2\sigma^2$. (b) Actual training data (1000 points). (c) Actual conditional probability $P(\mathbf{a} \in \text{class 1})$. Points falling in a lighter region are more likely to belong to class 1, darker to class 2. (d) Gaussian model estimated conditional probabilities. (e) Optimal decision boundary. Points appearing in white area are assigned to class 1, black to class 2. (f) Gaussian model decision boundary.

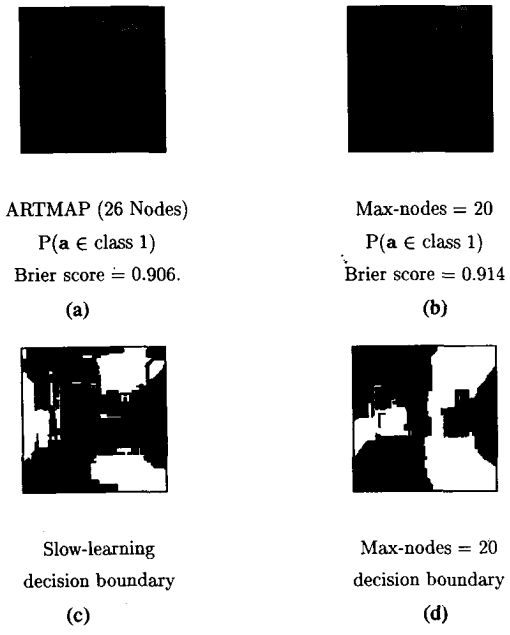

Fig. 5. Fuzzy ARTMAP model estimated conditional probabilities and decision boundary for six-Gaussians problem. (a) Slow-learning model estimated conditional probability, averaged over nine orderings of the training data. Map field learning rate $\beta_{ab} = 0.02$, map field matching criterion $\rho_{ab} = 0.75$. (b) Max-nodes = 20 model estimated conditional probability, averaged over nine orderings of the training data. $\beta_{ab} = 0.01, \rho_{ab} = 1$ until 20 ART_a nodes are committed, then $\rho_{ab} = 0$. (c) Slow-learning ARTMAP model decision boundary. (d) Max-nodes = 20 ARTMAP model decision boundary.

compression ratio of over 38 : 1. In max-nodes mode, the maximum number of categories was set to 20, for a data compression ratio of 50 : 1. Although this problem is more difficult than the two-Gaussian problem, fuzzy ARTMAP performed robustly, achieving Brier scores of 0.906 in slow-learning mode and 0.914 in max-nodes mode.

VII. Simulation: Noisy Nested Spirals

The probability estimation task presented in Fig. 6 is a variation of the nested spiral benchmark classification task described by Lang and Witbrock [8]. In their task, 97 input points belonging to class 1 fell along one spiral, and 97 points belonging to class 2 fell along a second, nested spiral. The noisy nested spiral task generates an input set from 194 Gaussian clusters, each centered at a point on one of the spirals. The white circles in Fig. 6(a) are centered at the means of the 97 Gaussians which make up class 1, and the black circles are centered at the means of the 97 Gaussians that make up class 2. Twenty patterns were drawn from each Gaussian, for a total of 1940 patterns belonging to each class [Fig. 6(b)]. Fig. 6(c) shows the actual probability that a pattern falling at each point in the unit square will belong to each of the two classes, as calculated using Bayes' rule. Fig. 6(d) shows the probability estimate of the two-Gaussian model. Fig. 6(e) and (f) compare the ideal classification and the two-Gaussian model classification, respectively.

Fig. 7(a) and (b) show the average probability estimate of the ARTMAP model in slow-learning mode and max-nodes mode, averaged over nine independent orderings of the training data. In max-nodes mode, the maximum number of nodes was set to 75, for a data compression ratio of over 50 : 1. The slow-learning mode created an average of 88 nodes, for a data compression ratio of almost 45 : 1. In both modes, ARTMAP correctly extracted the shape of the underlying spirals and assigned darker color to the upper left region and lighter color to the lower right. Fig. 7(c) shows the slow-learning decision boundary that results from assigning regions to the class with the higher estimated *a posteriori* probability. Fig. 7(d) shows the corresponding decision boundary for max-nodes mode. Note that in this example, the Brier score is an underestimate of fuzzy ARTMAP performance because it calculates pointwise errors that do not reflect the network's capacity to capture the fine structure and geometry of the nested spirals.

VIII. Discussion

In summary, the map field learning algorithms developed here expand the range of fuzzy ARTMAP applications by allowing the network to operate either as a classifier (in fast-learn mode) or as a probability estimator (in slow-learn or max-nodes mode). In each mode the system achieves a high degree of data compression and predictive accuracy.

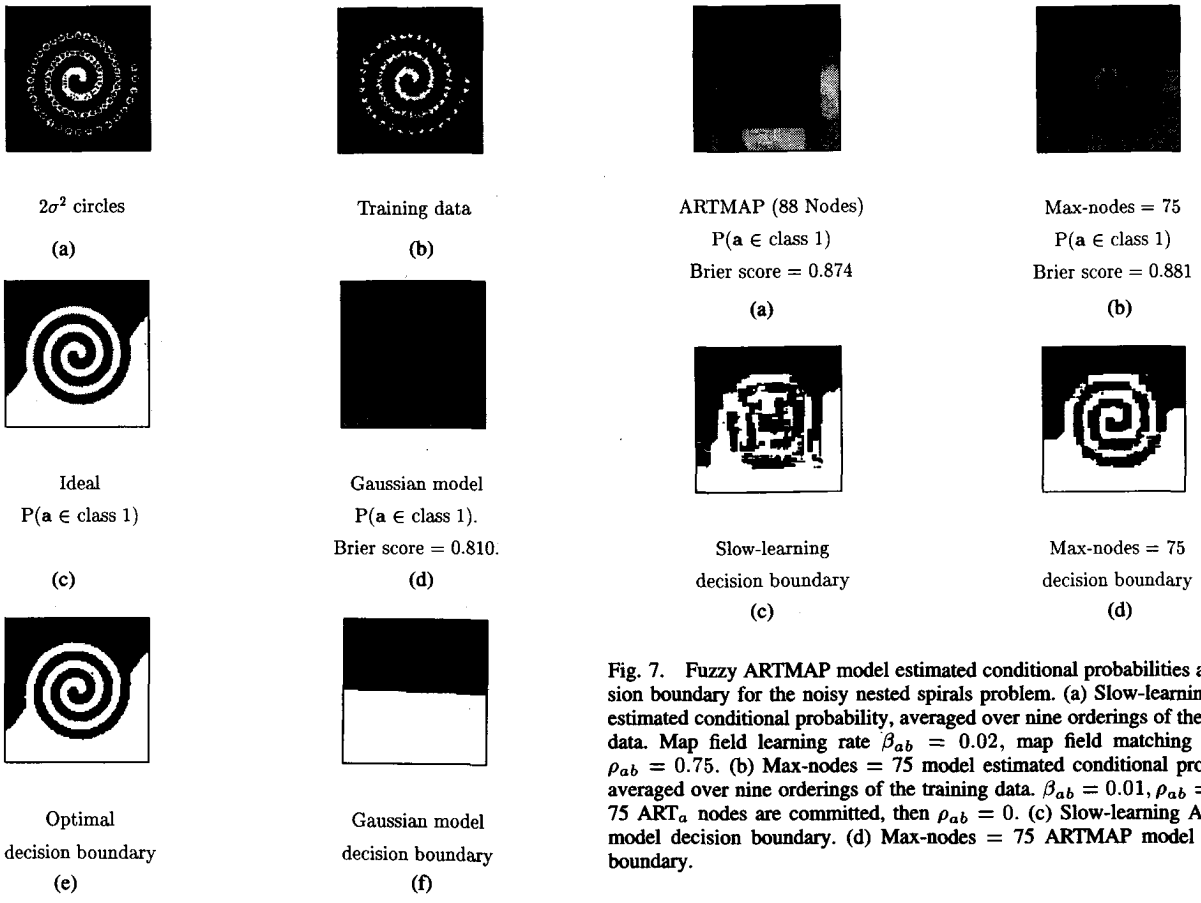

Fig. 6. Noisy nested spirals problem. (a) Circles around Gaussian means, with radius $2\sigma^2$. (b) Actual training data (3880 points). (c) Actual conditional probability $P(\boldsymbol{a} \in \text{class 1})$. Points falling in a lighter region are more likely to belong to class 1, darker to class 2. (d) Gaussian model estimated conditional probabilities. (e) Optimal decision boundary. Points appearing in white area are assigned to class 1, black to class 2. (f) Gaussian model decision boundary.

Fig. 7. Fuzzy ARTMAP model estimated conditional probabilities and decision boundary for the noisy nested spirals problem. (a) Slow-learning model estimated conditional probability, averaged over nine orderings of the training data. Map field learning rate $\beta_{ab} = 0.02$, map field matching criterion $\rho_{ab} = 0.75$. (b) Max-nodes = 75 model estimated conditional probability, averaged over nine orderings of the training data. $\beta_{ab} = 0.01, \rho_{ab} = 1$ until 75 ART_a nodes are committed, then $\rho_{ab} = 0$. (c) Slow-learning ARTMAP model decision boundary. (d) Max-nodes = 75 ARTMAP model decision boundary.

It is robust and is applicable to problems whose training input structures vary significantly in complexity. In all three simulation tasks (Sections V–VII), fuzzy ARTMAP correctly mapped the geometry of major regions of the input space that could be assigned, with high probability, to one of the output classes and achieved a high degree of data compression.

Fuzzy ARTMAP with slow learning offers solutions to problems inherent in many probability estimation applications. One such problem is the "curse of dimensionality." Fuzzy ARTMAP automatically selects input features needed to separate categories and thus can reduce the computational problems of high-dimensional input vectors. Another common problem is identifying how many data points are needed to yield an accurate probability estimate. In general, this is a difficult problem which depends on the particular application. Fuzzy ARTMAP can continue on-line learning after it has incorporated an initial training set. Thus, if the first data set is insufficient to generate an accurate enough probability estimate, additional data can be presented to the system for incremental learning, without having to retrain with the entire input set. Finally, because it does not depend on *a priori* knowledge of the data, fuzzy ARTMAP is especially useful when underlying input distributions are unknown or do not fit standard distribution patterns.

APPENDIX
SLOW-LEARNING EXAMPLE

The following example illustrates the steps of a slow learning simulation. Suppose input vectors $\boldsymbol{a}^{(1)}$ and $\boldsymbol{b}^{(1)}$ initially activate ART_a category J and ART_b category K, respectively. If the map field weight w_{JK}^{ab} projecting from ART_a category J to ART_b category K is smaller than the map field vigilance parameter, ρ_{ab}, then match tracking (Fig. 1) will cause an ART_a search, leading to a different active ART_a category. This reset-search-choice sequence will repeat until the map field weight w_{jK}^{ab} from an active ART_a category j to the correct ART_b category K is larger than ρ_{ab}. If no learned ART_a category is found to satisfy this condition, a previously uncommitted ART_a category is established. Then, weight w_{jK}^{ab} remains at its initial value of one, while all other weights $w_{jk}^{ab}(k \neq K)$ decay by an amount determined by the size of β_{ab}, by (1). The map field matching criterion is most easily satisfied when ρ_{ab} is small. Thus, setting ρ_{ab} to a low value will result in fewer ART_a nodes and greater code compression.

As soon as the weight w_{jK}^{ab} from the active ART_a category j to the correct ART_b category K is found to be greater than ρ_{ab}, all map field weights w_{jk}^{ab} are adjusted according to (1). One weight, w_{jK}^{ab}, from the active ART_a category node j to the chosen ART_b category node K, increases toward 1.0, while all other weights from the active ART_a node to the inactive ART_b nodes $(k \neq K)$ decay toward 0.0. Until w_{jk}^{ab} falls below ρ_{ab}, over multiple activations of the ART_a category j, the category's weight vector converges to a time-weighted average proportional to a degree of confidence in the prediction that

the corresponding ART_b category k will be correct given that the ART_a category j is chosen.

ACKNOWLEDGMENT

The authors wish to thank C. Y. Jefferson for her valuable assistance in the preparation of the manuscript.

REFERENCES

[1] 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 Trans. Neural Networks,* vol. 3, pp. 698–713, 1992.

[2] G. Carpenter, S. Grossberg, and J. Reynolds, "ARTMAP: Supervised real-time learning and classification of nonstationary data by a self-organizing neural network," *Neural Networks,* vol. 4, pp. 565–588, 1991.

[3] G. Carpenter, S. Grossberg, and K. Iizuka, "Comparative performance measures of fuzzy ARTMAP, learned vector quantization, and back propagation for handwritten character recognition," in *Proc. IJCNN-92,* vol. 1, 1992, pp. 794–799.

[4] P. Goodman, V. Kaburlasos, D. Egbert, G. Carpenter, S. Grossberg, J. Reynolds, K. Hammermeister, G. Marshall, and F. Grover, "Fuzzy ARTMAP neural network prediction of heart surgery mortality," in *Proc. Wang Conf. Neural Networks Learning, Recognition, Contr.,* 1992, p. 48.

[5] P. Goodman, V. Kaburlasos, D. Egbert, G. Carpenter, S. Grossberg, J. Reynolds, D. Rosen, and A. Hartz, "Fuzzy ARTMAP neural network compared to linear discriminant analysis prediction of the length of hospital stay in patients with pneumonia," in *Proc. IEEE Int. Conf. Syst., Man, Cybern.,* vol. 1, 1992, pp. 748–753.

[6] G. Carpenter and S. Grossberg, "A massively parallel architecture for a self-organizing neural pattern recognition machine," *Comput. Vision, Graphics, Image Process.,* vol. 37, pp. 54–115, 1987.

[7] G. Carpenter, S. Grossberg, and D. Rosen, "Fuzzy ART: Fast stable learning and categorization of analog patterns by an adaptive resonance system," *Neural Networks,* vol. 4, pp. 759–771, 1991.

[8] K. Lang and M. Witbrock, "Learning to tell two spirals apart," in *Proc. 1988 Connectionist Models Summer School,* 1989, pp. 52-59.

Gail A. Carpenter, for a photograph and biography, see p. 818 of the July 1995 issue of this TRANSACTIONS.

Stephen Grossberg received the Ph.D. degree in mathematics from Rockefeller University in 1967.

He is Wang Professor of Cognitive and Neural Systems and Professor of Mathematics, Psychology, and Biomedical Engineering at Boston University, MA, where he is the Founder and Director of the Center for Adaptive Systems, as well as the Founder and Chairman of the Department of Cognitive and Neural Systems.

Dr. Grossberg founded and was first President of the International Neural Network Society, and is co-Editor-in-Chief of the society's journal, *Neural Networks.* He was General Chairman of the first IEEE International Conference on Neural Networks and was awarded the 1991 IEEE Neural Network Pioneer prize. He pioneered the introduction and development of nonlinear real-time neural networks, including competitive learning and SOFM, ART, competitive winner-take-all nets; BCS, FCS, and FACADE vision nets; VITE, VITEWRITE, FLETE, VAM, and DIRECT sensory-motor control nets; STORE temporal order nets; VIEWNET 3-D object recognition nets; SPINET, ARTSTREAM, ARTPHONE, and PHONET auditory and speech nets; and START spectrally timed reinforcement learning nets.

John H. Reynolds (M'95) received the Ph.D. degree in 1993 from the Boston University, MA, Department of Cognitive and Neural Systems, where his dissertation research included development and analysis of the ARTMAP and fuzzy ARTMAP neural network architectures for supervised learning and prediction.

He is currently a Fellow in the Laboratory of Neuropsychology at the National Institutes of Health, where he uses a combination of neurophysiological and neural modeling techniques to investigate the cortical and subcortical mechanisms of spatially selective visual attention.