Probabilistic Fuzzy ARTMAP : An Autonomous Neural Network Architecture For Bayesian Probability Estimation

Chee Peng Lim and Robert F. Harrison

Department of Automatic Control and Systems Engineering
The University of Sheffield
PO Box 600, Mappin Street
Sheffield, S1 4DU, UK
Electronic mail: c.lim@sheffield.ac.uk, r.f.harrison@sheffield.ac.uk

Abstract

A hybrid utilisation of the Fuzzy ARTMAP (FAM) neural network and the Probabilistic Neural Network (PNN) is proposed for on-line learning and prediction tasks. FAM is used as an underlying clustering algorithm to classify the input patterns into different recognition categories during the learning phase. Subsequently, a non-parametric probability estimation procedure in accordance with the PNN paradigm is employed during the prediction phase. This hybrid approach realises an incremental learning network with implementation of the Bayes strategy for on-line applications. The effectiveness of this network is assessed with statistical classification problems in both stationary and non-stationary environments. Simulation studies illustrate that the network is capable of asymptotically approaching the Bayes optimal classification rates.

Research Report No. 566
February 1995

1 INTRODUCTION

Autonomous knowledge acquisition has always been a major research area in machine learning. Recently, researchers have shown increasing interest in using neural networks as the core of learning systems. This is mainly because neural networks can acquire knowledge from exemplars and then generalise the knowledge to cover the entire problem domain. However, when using feedforward neural networks, such as the Multi-Layer Perceptron (MLP) networks and the Radial Basis Function (RBF) networks, one will have to rely upon some heuristics to select the "optimum" network size and parameters (Lippmann (1)). Besides, these networks are usually *static* after training. If the problem domain is dynamic and non-stationary, re-training the feedforward networks with newly available information is necessary as it is very difficult to allow them to continue learning in perpetuity.

In contrast, the Adaptive Resonance Theory (ART) (Carpenter & Grossberg (2)) family of neural networks are developed to overcome the so-called stability-plasticity dilemma (2). These networks are able to learn in changing environments where new information can be accommodated to its knowledge base without corrupting previously learned information. On the other hand, the Probabilistic Neural Network (PNN) (Specht (3), (4)) also has autonomous learning properties similar to those of ART. In this paper, we propose a hybrid utilisation of Fuzzy ARTMAP (FAM) (Carpenter et al (5)), a supervised ART network, with the PNN for on-line learning and classification tasks, and compare the results with the Bayes optimal rates.

2 FUZZY ARTMAP AND THE PROBABILISTIC NEURAL NETWORK

FAM is a variant of the supervised ARTMAP (Carpenter et al (6)) architecture in which fuzzy set theory is incorporated to govern the dynamics of ARTMAP. Fig. 1 shows a schematic diagram of the FAM network for binary classification tasks. In general, FAM consists of two identical fuzzy ART (Carpenter et al (7)) modules, ART_a and ART_b, linked by a map field, F^{ab}. Each ART module has two layers of nodes: $F_{1a/1b}$ is the input layer; and $F_{2a/2b}$ is a dynamic layer where every single node encodes a prototype pattern of the input samples. The number of nodes in F_{2a} can be increased when necessary.

During supervised learning, ART_a receives an input pattern and ART_b receives its target output. In ART_a (as well as in ART_b), in order to avoid the category proliferation problem (7), the input pattern $\boldsymbol{a}$ is complement-coded in F_{1a}, i.e. $A = (\boldsymbol{a}, 1-\boldsymbol{a})$, before it is transmitted to F_{2a}. F_{2a} is a winner-take-all competitive layer where a choice function is used to measure the response of each prototypical node as follows:

$$\frac{|A \wedge w_{a-j}|}{\alpha_a + |w_{a-j}|} \tag{1}$$

where w_{a-j} is the weight vector of the jth F_{2a} node; and α_a is the choice parameter of ART_a (5). The fuzzy "and" operator $(\wedge)$ and the norm $|\,.\,|$ are defined as: $(x \wedge y)_i \equiv \min(x_i, y_i)$ and $|x| \equiv \sum_i |x_i|$ (Zadeh (8)).

The maximally responsive node is selected as the winner while all other nodes are shut down. The winning node then sends its weight vector (the prototype pattern) to F_{1a}. A vigilance test is performed to check the similarity between the prototype pattern and the input pattern, i.e.

$$\frac{|A \wedge w_{a-J}|}{|A|} \geq \rho_a \tag{2}$$

where ρ_a is the vigilance parameter of ART_a; and w_{a-J} is the Jth wining node in F_{2a}. If this test is satisfied, resonance occurs and learning takes place. However, if it fails, the winning node will be inhibited and the input pattern will be re-transmitted to F_{2a} to search for a new winner which fulfils the vigilance test. If such a node does not exist, a new node is recruited to code the input pattern.

To impose supervision, the map field associates the winner in F_{2a} with the target winner in F_{2b}. This association is permanent so that a target output can be recalled during the prediction phase.

2.1 Modified Fuzzy ARTMAP

According to our previous work (Lim & Harrison (9)), we indicate that FAM is unable to establish one-to-many mappings, i.e. forming an association from an F_{2a} prototypical node to more than one F_{2b} target output via the map field. This mapping is crucial in statistical pattern classification tasks where overlapping regions can occur in the input space in which a particular cluster may belong to more than one class subject to different probabilities of class membership.

In FAM, ρ_a is dynamically increased during *match-tracking* ((5), (6)) when a prediction is rejected by the map field. A search is initiated which leads to the selection of a new F_{2a} prototypical node that satisfied equation (2), or to the shut-down of the F_{2a} layer if no

such node exists (5).

We therefore propose a constraint on ρ_a, during match-tracking, as follows:

$$0 \leq \rho_a \leq \min\left(1, \frac{|A \wedge w_{a-J}|}{|A|}\right) \quad (3)$$

where A is the current input vector to ART_a in complement-coded format; and w_{a-J} is the winning Jth node in F_{2a}. The effect of the constraint is to recruit a new node in F_{2a} to code the input pattern instead of just ignoring it as in the original algorithm, thus having two similar prototypical nodes to map to different target outputs. In addition, a frequency measure scheme which records the number of correct predictions of nodes in F_{2a} is also introduced and this information is used to facilitate the selection of the winning node. Two variants of the frequency measure scheme have been proposed (9): INC only and INC/DEC based on the reward and reward/penalty rationale. The INC method records correct predictions only whereas the INC/DEC method increases or decreases the frequency counts corresponding to correct or incorrect predictions respectively.

We assessed the applicability of Modified FAM with a common classification task: the separation of two multi-dimensional sources. Two classes of Gaussian distributed, continuous-valued random variables with different source separations (means), variances and prior probabilities were generated. Our studies have found that Modified FAM is able closely to approximate the Bayes optimal error rates in various configurations of the problem in both stationary and non-stationary environments, whereas FAM is not. Nevertheless, we have also realised that it has some difficulties with real-valued input patterns. This is because Modified FAM fails to devise an efficient and flexible technique in: (i) forming non-linear decision boundaries in multi-dimensional cases; and (ii) tracking the changes of the optimal decision boundaries in non-stationary environments.

2.2 Probabilistic Neural Network

The PNN is a neural network model that directly implements the Bayes strategy for pattern classification in its learning paradigm. It learns instantaneously in one-pass through the data samples and is able to formulate complex decision boundaries which approximate the Bayes optimal limits. Besides, the decision boundaries can be modified on-line when new data is available without having to re-train the network. However, the key feature of the PNN is its ability to estimate the probability density functions by using the Parzen window (Parzen (10)) based on the data samples, i.e. a non-parametric density estimation procedure.

Fig. 2 depicts a schematic diagram of the PNN for binary classification task. In the basic PNN, the input pattern, x, is first fanned-out to the pattern layer where each pattern unit forms a dot-product of the input and weight patterns. The dot-product is transformed by an activation function in accordance with the Parzen kernel estimator ((3), (4)). The summation units then add all the outputs of pattern units corresponding to each category to give estimates of probability density functions ($P(x|A)/P(x|B)$). For classification problems, these estimates can be weighted by their respective *a priori* probabilities ($P(A)/P(B)$). This enables the output unit to calculate the *a posteriori* probability of x belonging to a particular category according to Bayesian decision criterion, e.g. $P(A|x) = P(x|A)P(A)/P(x)$.

Learning in the PNN is accomplished by generating a new pattern unit for each input pattern and encoding it as the weight pattern. The pattern unit is then linked to the summation unit of the class of the current input pattern. This process is non-iterative and can be implemented on-line. Note that in addition to kernels using dot-product inputs, many alternative forms of estimators have also been described by Specht (4), (12).

2.3 Probabilistic Fuzzy ARTMAP

Our studies have discovered a close similarity in the network connections between FAM and PNN as shown in Fig. 1 and 2. The F_{1a} and F_{2a} layers correspond to the input and pattern layers whereas the map field layer (F_{ab}) corresponds to the summation layer. In essence, in one-from-N classification, each node in F_{2a} is permanently associated with only one node in F_{ab} through the map field weights (w_{ab}), which is then linked to the target output in F_{2b}. Thus, the F_{ab} nodes can be used to sum up outputs from all the F_{2a} nodes corresponding to a particular target category, taking the role of the summation units in the PNN.

In fact, the major drawback of the PNN is to recruit a new pattern unit for every input pattern which leads to an explosive number of pattern units when large or unbounded data sets are available. As suggested in the literature (Burrascano (11), Specht (12), Musavi (13)), this problem can be remedied by using a clustering technique to reduce the number of pattern units required so that each pattern unit represents a cluster of input patterns.

According to Lippmann (1), the learning procedure of ART is actually similar to the so-called sequential leader clustering algorithm. In view of the suitability of the learning methodology and the likeness of the network connections, FAM provides a natural platform where the two networks can be incorporated. As a result, we propose a Probabilistic FAM for on-line applications where FAM is used as the underlying clustering algorithm during the learning phase. Conversely, the PNN is used to give a Bayesian probability estimation of the outputs during the prediction phase.

In FAM, only one of the elements in w_{ab} is unity which indicates the link from an F_{2a} node to the appropriate F_{ab} node. However, in order to accommodate clustering, the link in w_{ab} should be incremented by one if the F_{2a} node successfully classifies an input pattern. Thus, the outputs can then be weighted by w_{ab} to represent the strength of different cluster prototypes. Besides, summing the links in w_{ab} provides information of the prior class probabilities.

In summary, the algorithm of Probabilistic FAM is as follows:

(a) Learning Phase

(1) Feedforward the input pattern and determine the winner according to the choice function of equation (1).
(2) Feedback the prototype pattern and perform the vigilance test as in equation (2).
(3) If there is no match, trigger the search cycle and Goto (1).
Else Goto (4).
(4) Update the weights of the winning node. Increment the link in w_{ab} by one.

(b) Prediction Phase

(1) Feedforward the input pattern and compute the inputs to the kernel estimators.
(2) Perform the activation functions upon the outputs from the pattern units.
(3) Sum the kernel estimations weighted by w_{ab} for each category.
(4) Select the highest *a posteriori* estimate and predict the target output in ART_b.

Note that the effect of equation (3) should be taken into consideration in Probabilistic FAM for implementing one-to-many mappings. However, the frequency measure information has been encoded in w_{ab}.

3 SIMULATION STUDIES

To demonstrate the capabilities of Probabilistic FAM, we re-investigate the two problems addressed with Modified FAM, i.e. multi-dimensional inputs and non-stationary cases. Both FAM and Probabilistic FAM use the on-line approach operating in the conservative mode (the choice parameter $\alpha \rightarrow 0$) (5). The on-line operational cycle proceeds as follows: an input pattern is first presented to ART_a with its target output to ART_b. Second, a predicted class is sent from F_{2a} winner to F_{2b}. Then, the prediction is compared with the actual class and the outcome gives a classification result (prediction phase). Finally, learning ensues to associate the input pattern with its target class (learning phase).

In all experiments, the input patterns were in complement-coded format and the kernel estimators used were the "city-block" distance metric (12) to obviate normalisation of input patterns to unit length. Some of the important network parameters used were: the vigilance parameter of ART_a, $\rho_a = 0.5$; the learning rate parameter, $\beta = 1$ (fast learning); and the smoothing parameter, $\sigma = 0.1$.

3.1 Stationary on-line learning and classification

Here, two sources of continuous-valued, Gaussian-distributed random variables were generated with fixed prior probabilities ($P\{c_1\} = P\{c_2\} = 0.5$) and variances ($\sigma_1^2 = \sigma_2^2 = 1.0$). Class 1 and class 2 were represented by multivariate normal distributions with mean vectors $\mu_1 = (-1.0,0,\ldots,0)$ and $\mu_2 = (1.0,0,\ldots,0)$. The dimension of the input samples was increased from two to six. Although the data parameters are time-invariant, tackling the task on-line is in fact a non-stationary process, owing to the build-up of templates—the so-called finite-operating-time problem. In each simulation, 5000 input samples were generated and a 1000-sample window was applied for calculating the accuracy, e.g. the accuracy at sample 2000 was the percentage of correct predictions from trials 1001-2000.

Table 1 shows the average results of 5 runs and their standard deviations at the end of the sample presentation. Fig. 3(i) depicts a typical on-line accuracy plot against increasing number of samples for six-dimensional input samples. From Table 1, Probabilistic FAM shows an improvement of at least 15% over FAM with smaller standard deviations in all cases. Although the standard deviations are estimated from a small sample size (5 runs), it indicates a dispersion of the results across the averages. Note that

the results can sometimes exceed the optimum Bayes limits owing to the use of the window method in calculating the accuracy.

3.2 Non-stationary on-line learning and classification

In this experiment, we investigate the classification abilities of FAM and Probabilistic FAM in non-stationary environments. Two properties were evaluated, *viz.* the ability to track non-stationarity and the approximation to the Bayes limits. Here, two classes of single-dimensional Gaussian distributed random variables were generated with a sample size of 25000. But the data statistics were subject to step changes at each 5000 samples. This simulates a severe non-stationary scenario as one might expect to experience a gradual change of data statistics to enter the environment rather than a step change. In order to track non-stationarity, a forgetting factor is introduced in the operation of Probabilistic FAM. A factor of $1/k$ was applied to the links in w_{ab} where k was the window length of the on-line accuracy.

Table 2 shows the parameters used to generate the input samples. Fig. 3(ii) depicts the average results of 5 runs with some indications of their standard deviations. From Fig. 3(ii), it is clear that Probabilistic FAM not only successfully tracks non-stationarity in the data environment but is capable of asymptotically approaching the Bayes limit.

4 DISCUSSION

Studies of the ability of FAM in probability estimation have been reported in Carpenter et al (14). They investigated FAM in two modes, "slow-learning" and "mix-nodes", where different learning strategies were adopted for probability estimation of two noisy, nested spirals. Our work here investigates the effectiveness of Probabilistic FAM operating in fast-learning, on-line mode in binary classification tasks.

The two simulations above demonstrate that Probabilistic FAM is able to overcome the difficulties of Modified FAM in handling multi-dimensional input patterns and non-stationary environments. Unlike Modified FAM which uses the frequency measure scheme to formulate the separating decision boundaries, Probabilistic FAM forms non-linear decision boundaries that approximate the Bayes-optimal by using the Parzen window approach. A maximum *a posteriori* decision criterion is then directly applied as part of its implementation to select the most probable output. In addition to providing probabilistic outputs, the decision process can also be adjusted and weighted by different risk or loss factors. More importantly, the Bayes strategy is implemented on-line without having to re-train the network when auxiliary information becomes available.

Nevertheless, there are a few difficulties associated with the combination of FAM and PNN. To obviate the need of complement-coding, one can normalise the input patterns to unit length and employ the dot-product kernel estimators. However, owing to the fuzzy learning rules of FAM in complement-coding and fast-learning mode, the prototypical patterns are represented by vertices of the hyper-rectangles that define the cluster boundaries (5). In other words, the patterns encoded in F_{2a} nodes do not reflect the centroids of input clusters. Another problem is the need to select a smoothing parameter for the kernel estimators and one often has to resort to some heuristic in determining the "optimum" value. To overcome this problem, Musavi et al (13) have proposed the construction of the covariance matrices of Gaussian kernel estimators by using the Gram-Schmidt orthogonalisation process. Besides, the convergence of the Parzen window estimation procedure is guaranteed only when the data samples extend to infinity.

5 CONCLUSIONS

A hybrid network which incorporates the FAM and PNN architectures is presented. The advantages of Probabilistic FAM are two-fold: (i) it provides a probabilistic interpretation of predicted outputs according to the Bayes strategy by using the PNN as the underlying estimation mechanism; and (ii) it reduces the number of pattern units required by using FAM as the underlying clustering mechanism. Simulation studies demonstrate that Probabilistic FAM is able to classify statistical patterns on-line in both stationary and non-stationary environments and simultaneously achieve the Bayes optimal classification rates. Current work suggests that further improvements are necessary, particularly in relation to determining the smoothing parameter and cluster centres autonomously and on-line.

REFERENCES

1. Lippmann, R.P., 1987, "An Introduction to Computing with Neural Nets", IEEE ASSP Magazine, 4-22.

2. Carpenter, G.A., Grossberg, S., 1987, "A Massively Parallel Architecture for a Self-Organizing Neural Pattern Recognition Machine", Computer Vision, Graphics and Image Processing, 37, 54-115.

3 Specht, D.F., 1988, "Probabilistic Neural Network for classification, mapping or associative memory", Proc. of IEEE Int. Conf. on Neural Networks, I, 525-532.

4 Specht, D.F., 1990, "Probabilistic Neural Networks", Neural Networks, 3, 109-118.

5 Carpenter, G.A., Grossberg, S., Markuzon, N., Reynolds, J.H., Rosen, D.B., (1992), "Fuzzy ARTMAP: A Neural Network Architecture for Incremental Supervised Learning of Analog Multidimensional Maps", IEEE Trans. on Neural Networks, 3, 698-712.

6. Carpenter, G.A., Grossberg, S., Reynolds, J.H. (1991), "ARTMAP: Supervised Real-Time Learning and Classification of Nonstationary Data by a Self-Organizing Neural Network", Neural Networks, 4, 565-588.

7. Carpenter, G.A., Grossberg, S., Rosen, D.B., (1991), "Fuzzy ART : Fast Stable Learning and Categorization of Analog Patterns by an Adaptive Resonance System", Neural Networks, 4, 759-771.

8 Zadeh, L.A., 1965, "Fuzzy Sets", Inform. Control, 8, 338-353.

9. Lim, C.P., Harrison, R.F., "Modified Fuzzy ARTMAP Approaches Bayes Optimal Classification Rates: An Empirical Demonstration", To appear in Neural Networks.

10. Parzen, E., 1962, "On Estimation of a Probability Density Function and Mode", Annals of Mathematical Statistics, 33, 1065-1076.

11. Burrascano, P., 1991, "Learning Vector Quantization for the Probabilistic Neural Network", IEEE Trans. on Neural Networks, 2, 458-461.

12. Specht, D.F., 1992, "Enhancements to Probabilistic Neural Networks", Proc. of Int. Joint Conf. on Neural Networks, I, I761-I768.

13. Musavi, M.T., Kalantri, K., Ahmed, W., Chan, K.H., 1993, "A Minimum Error Neural Network (MNN)", Neural Networks, 6, 397-407.

14. Carpenter, G.A., Grossberg, S., Reynolds, J.H., 1993, "Fuzzy ARTMAP, Slow Learning and Probability Estimation", Proc. of World Congress of Neural Networks, II, pp 26-30.

TABLE 1 Average results of 5 runs at the end of the data presentation in stationary environments

Input Dimension	Bayes Limit	Probabilistic FAM		Fuzzy ARTMAP	
	Accuracy (%)	Accuracy (%)	Standard Deviation	Accuracy (%)	Standard Deviation
2	84.13	84.4	1.1	69.1	1.8
3	84.13	84.4	0.5	69.3	1.1
4	84.13	84.7	0.7	69.6	2.0
5	84.13	84.1	0.7	67.5	0.9
6	84.13	83.4	0.8	67.4	2.6

TABLE 2 The data statistics are changed every 5000 samples to simulate non-stationary environments

Samples	Class 1			Class 2		
	Mean	Standard Deviation	Prior Probability	Mean	Standard Deviation	Prior Probability
1-5000	-0.5	1.0	0.1	0.5	1.0	0.9
5001-10000	-0.5	2.5	0.5	0.5	2.5	0.5
10001-15000	-0.5	1.0	0.5	0.5	1.0	0.5
15001-20000	-0.5	0.5	0.5	0.5	4.5	0.5
20001-25000	-0.5	1.0	0.5	0.5	1.0	0.5

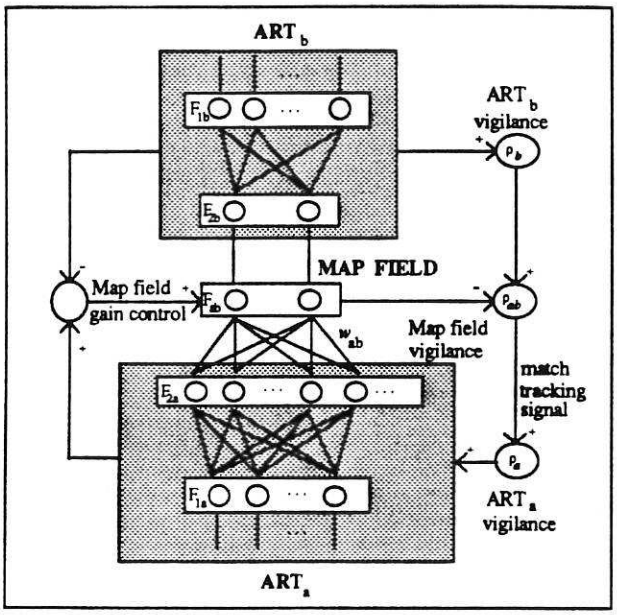


Fig. 1. A schematic diagram of Fuzzy ARTMAP

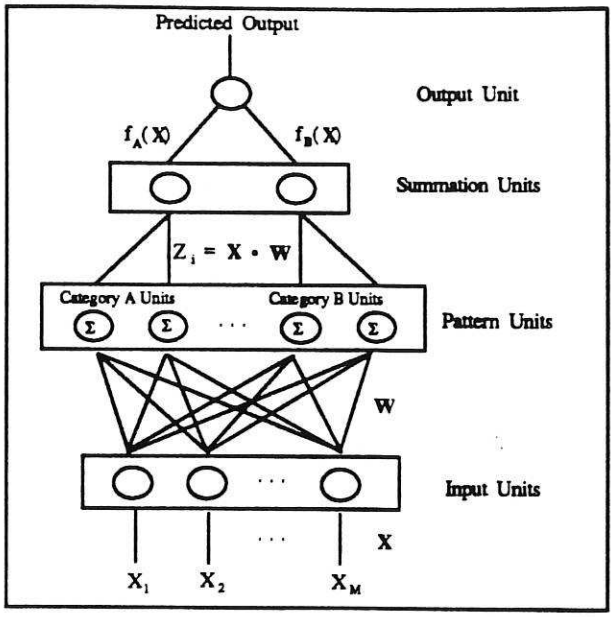
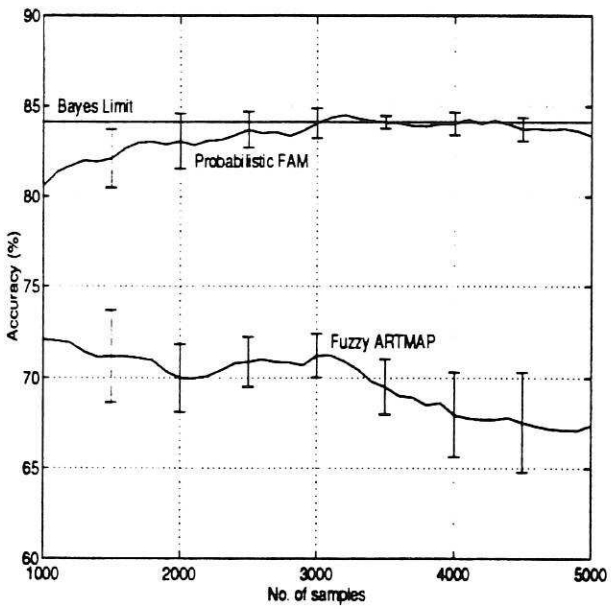
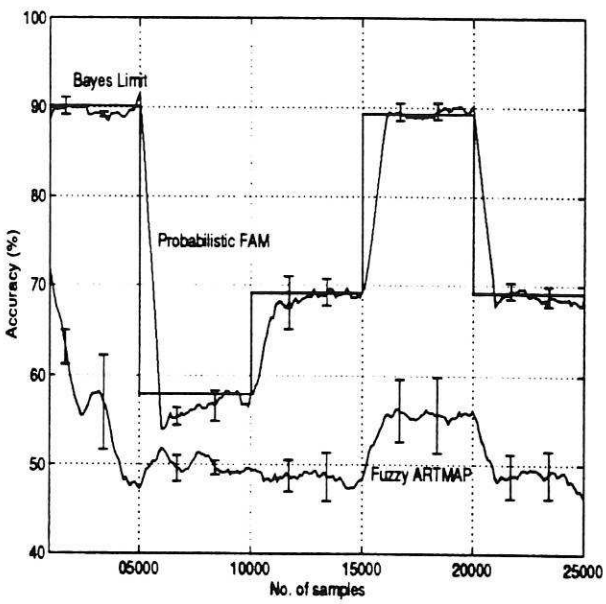


Fig. 2 A schematic diagram of Probabilistic Neural Network



(i) six-dimensional input samples



(ii) single-dimensional input samples

Fig. 3 Comparison between Fuzzy ARTMAP and Probabilistic FAM in (i) stationary, and (ii) non-stationary environments