GENERALIZATION PERFORMANCE ANALYSIS BETWEEN FUZZY ARTMAP AND GAUSSIAN ARTMAP NEURAL NETWORK

Shahrul Nizam Yaakob[1]***, Puteh Saad***[2]

[1] Artificial Intelligence and Software Engineering Research Lab
School of Computer and Communication Engineering, Universiti Malaysia Perlis (UNIMAP)
02600 Jejawi, Perlis. Email:ysnizam@gmail.com
[2] Department of Software Engineering
Faculty of Computer Science and Information Systems
81310 UTM Skudai, Johor, Malaysia. Email:puteh@utm.my

ABSTRACT

This paper examines the generalization characteristic of Gaussian ARTMAP (GAM) neural network in classification tasks. GAM performance for classification during training and testing is evaluated using the k-folds cross validation technique. A comparison is also done between GAM and Fuzzy ARTMAP (FAM) neural network. It is found that GAM performs better (98-99%) when compared to FAM (79-82%) using two different types of dataset. The difference between GAM and FAM is that input data to be to classified using FAM must be normalized in prior. Hence, three different normalization techniques are examined namely; unit range (UR), improved unit range (IUR) and improve linear scaling (ILS). This paper also proposes an alternative technique to search the best value for gamma γ parameter of GAM neural network, known as gamma threshold. A small number of training required for GAM also shows that its fundamental architecture retain the attractive parallel computing and fast learning properties of FAM.

Keywords: ***Gaussian ARTMAP (GAM), Fuzzy ARTMAP (FAM), gamma threshold.***

1.0 INTRODUCTION

There are numerous technical applications that require the usage of artificial neural network (ANN). An example of the ANN is the supervised ARTMAP neural networks. The history of ARTMAP started in 1992 when Carpenter, G.A. *et al*. [6,7] introduced the Fuzzy ARTMAP neural network that is suitable for classification of both binary and analog input pattern. One of the most important features of ARTMAP is the ability of its algorithm to classify the untrained input patterns to the appropriate classes. However, for the past years other researchers have tried to improve the fundamental theory of FAM by introducing various techniques with brilliant concepts. For instance, Williamsons, J.R. created a new version of FAM that is known as Gaussian ARTMAP (GAM). In his paper [18], he had demonstrated that GAM outperformed its counterparts FAM in classification tasks especially with the noisy input pattern.

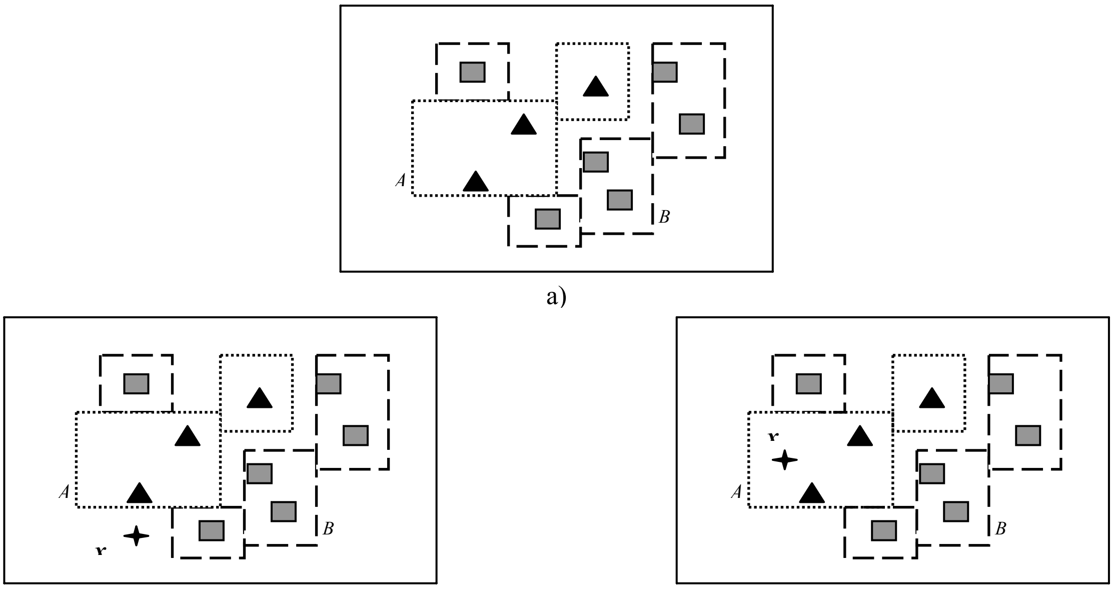
In order to enhance the generalization performance of GAM, developers need to set the value of parameters required for the classifiers. This step is important since the exact initial value for the parameters would increase the performance of the classifier. The characteristics of good classifiers are that they should be independent from tuning parameters characteristic [3] or in other word should have less parameter to be initialized by the developers. We observe that none highlighted the best technique to choose the suitable γ parameter of GAM. Only Williamsons advised to start training with a large γ and decrease γ as training progresses. He also said that the value of γ varies with different type of application dataset. However, in this paper we introduce a technique to deliberately search the best possible value for the gamma parameter of GAM γ. In addition, we also demonstrate that GAM would have better generalization capability when the suitable γ value is adopted.

The organization of this paper is as follows: In Section 2 we discuss in general about the generalization concept which is the main purpose of this paper. Section 3 explains about GAM and FAM algorithm respectively. Section 4 focuses on important elements in this work such as the dataset used, cross validation and normalization techniques (for FAM only) applied. Section 5 contains the result for both FAM and GAM in terms of their parameters initialization or cross validation results. Finally, Section 6 provides the conclusions and directions for future research.

2.0 GENERALIZATION PERFORMANCE

Basically, generalization refers to the capability of each ARTMAP neural networks to identify and recognize the untrained or unknown tested input vector with the correct output answer. Fig. 1 describes an example of how the classification and prediction process will help the ARTMAP neural networks to generate higher generalization performance. Assuming we have nine input pattern that represent two different group of data which is *A* and *B* after the training process was done. Let us say that we have a new input pattern *x* as shown in Fig. 1b and 1c. Therefore, an excellent neural network is capable of predicting which class is suitable to represent this input (e.g. either in group *A* or *B*).

However, from the supervised ARTMAP point of view this untrained input pattern can be described into two main conditions. First, if the input is inside the category space which is already learnt by the ARTMAP neural network (Fig. 1c). Secondly, if the input is initiated outside the category region generated in the recognition layer (Fig. 1b). A supervised ARTMAP neural network with good generalization characteristic should be able to cluster the input for both conditions into the targeted category. In this work we adopt the k-folds cross validation techniques to generate both types of circumstances. Therefore, we can evaluate the generalization performance for each classifier for a given dataset (e.g. in this paper we compare the performance of GAM and FAM).



a)

b)

c)

Fig. 1: Generalization measurement a) Six categories in the recognition layer for two different classes *A* and *B*. b) The untrained input pattern is outside the categories. c) The untrained input pattern within the categories.

3.0 SUPERVISED ARTMAP NEURAL NETWORK

Fuzzy ARTMAP (FAM) is a supervised neural network developed by Carpenter, G.A. et al. [6, 7]. FAM incorporates four basic combinations that are pair of Fuzzy ART modules ART_a and ART_b, Associate learning network, internal controller module and a mapfield F^{ab}. The details of this neural network are described in [6] for FAM while [18] for GAM.

4.0 EXPERIMENTAL WORK

In this section, we discuss all the important aspects that have been carried out in the implementation of this work. In order to examine the performance of GAM and its comparative counterpart FAM, we use an application data as the input pattern. Furthermore, the training and testing process is based on the cross validation technique.

4.1 Dataset

A comparative study between FAM and GAM neural network performance is conducted based on two different datasets. These datasets have been generated from the binary images that represent an insect using dissimilar features extraction techniques which are Geometric Legendre Moment Invariant (LMI) and Krawtchouck Moment Invariant (KMI). Each dataset consists of 240 feature vectors that represent 20 different insect images.

4.2 Cross Validation

In order to find the generalization performance of FAM and GAM in conjunction with different types of features extraction, the method of k-fold cross validation is used. In this work the available data *G* is split into *k* mutually exclusive subsets, designated as G_1, G_2, G_3,….G_k of equal size. The classifier is than trained and tested *k* – times. The value of *k* is set to be 4. The cross validation estimate is defined as the number of correct classification divided by the number of data points in the set *G*. The percentage of correct classification (PCC_k) is given by (1) and the number of correct classification (NCC_k) is given by (2). $\sigma(x,y)_t=1$ if the testing vector is true, while $\sigma(x,y)_t=0$ if otherwise and *n* refers to the number of data tested.

$$PCC_k = 100\frac{1}{G}\sum_{k=1}^{4} NCC_k \qquad (1)$$

$$NCC_k = \sum_{t=1}^{n} \sigma(x,y)_t \qquad (2)$$

4.3 Normalization Techniques for FAM

Since the value of FAM input range from zero (0) to one (1) and it is found that most of the result produced in stage 1 are not most on the particular range, therefore the normalization process are required. Thus, three (3) types of different normalizations techniques have been selected for this purpose. They are the simple unit range (SUR), improve unit range (IUR) and improved linear scaling (ILS). Equation (3), (4) and (5) show the computation needed for SUR, IUR and ILS respectively. The best technique adopted is based on the highest classification rate produced by FAM neural network. With reference to the equations, X' refers to the new value of feature, *X* in each dimension after the normalization process.

$$X' = \frac{X - X_{\min}}{X_{\max} - X_{\min}} \qquad (3)$$

$$X' = 0.8 * \left(\frac{X - X_{\min}}{X_{\max} - X_{\min}} \right) + 0.1 \qquad (4)$$

$$X' = \left(\frac{X - \mu}{3\sigma} + 1 \right) \div 2 \qquad (5)$$

Where: $\mu = \Sigma X_n \div n \qquad \sigma = \sqrt{\Sigma\left(X_n - \mu\right)^2 \div (n-1)}$

Furthermore $X_{\max}$ and $X_{\min}$ refer to the maximum and minimum features value respectively. For the ILS computation, μ refers to the *min* of the feature and σ is used for the standard deviation of the features in the same dimension. The dimensions for each vector are defined as *n*.

5.0 RESULTS AND DISCUSSION

In this section we discuss both neural networks with their parameters initialization. Then, we continue with the main cross validation result.

5.1 FAM Parameters Initialization

Like any other neural networks, FAM also has three (3) important parameters to be initialized, namely the learning parameter, choice and vigilance parameter. Table 1 describes the value chosen for all the parameters. The value for vigilance parameter of *ARTa* increases as training progress, while choice parameter can be at any number except zero (0). Since the vigilance will automatically increases, thus the starting value which is also known as the baseline vigilance is set to 0.5. One of the advantages of FAM is that it converges faster than other types of neural networks. FAM also can achieve high classification rates by using fast learning mode (i.e. by initializing the learning parameter to one (1) as demonstrated by Lopes, M.L.M. et al. [12] where he applied FAM in electric load forecasting. It is found that FAM requires only three (3) epochs training for most of the moment data used as shown in Table 2. Nevertheless, different types of moment and normalization techniques applied generate different value for vigilance and the number of categories in *ARTa*.

Table 1: Parameter setting for FAM

Parameter	Value
Baseline Vigilance, $\bar{\rho}$	0.5
Vigilance ARTa, ρ_a	1
Learning, β	1
Choice, α	0.01

Table 2: FAM training with β equal to one (1)

		Unit Range			**Improve unit range**			**Improve linear scaling**		
	e	*ρA*	*nA*	*NCC*	*ρA*	*nA*	*NCC*	*ρA*	*nA*	*NCC*
LMI	1	0.96	26	154	0.96	26	154	0.97	26	155
	2	0.96	40	170	0.97	40	170	0.97	38	240
	3	0.96	43	240	0.97	43	240	0.97	38	240
	4	0.96	43	240	0.97	43	240	0.97	38	240
	5	0.96	43	240	0.97	43	240	0.97	38	240
KM	1	0.98	22	175	0.98	22	175	0.98	22	175
	2	0.98	35	240	0.98	35	240	0.98	33	240
	3	0.98	35	240	0.98	35	240	0.98	33	240
	4	0.98	35	240	0.98	35	240	0.98	33	240
	5	0.98	35	240	0.98	35	240	0.98	33	240

Note: *NCC* = number of correct classification *e* = epoch
nA=No. of *ARTa* category

At this stage, we found that UR generates lower vigilance compared to IUR but the number of categories in *ARTa* is still the same. This is because IUR actually decreases the feature space available between 0.1 and 0.9 as shown in equation (4). Meanwhile, UR widens the data between 0 and 1 (Fig. 2a). These circumstances cause different groups of data closely tight together. Fig. 2 illustrates a situation when the vigilance value will increase in two different groups of data *A* and *B* in the same feature space *ξ*. Nevertheless, KMI, with ILS normalization technique, demonstrates the highest classification of insect images compared to other methods. Even though KMI produces the lowest categories number in *ARTa*, LMI on the other hand generates the lowest value for vigilance parameter.

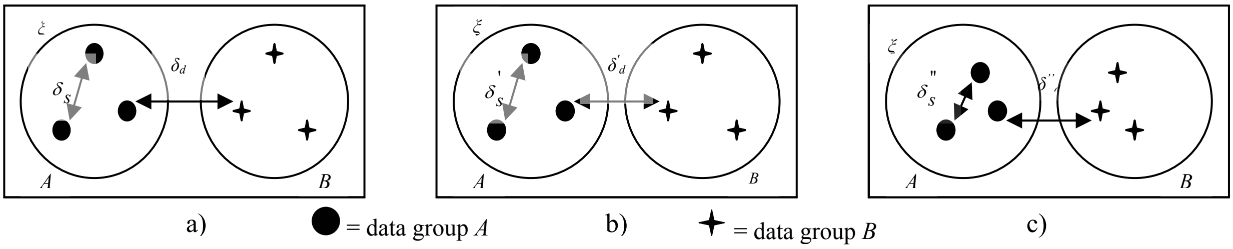


Fig. 2: Comparisons among normalization techniques a) UR, b) IUR and c) ILS

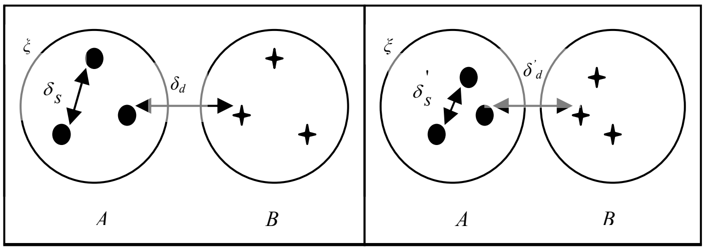
5.2 FAM Cross Validation Result

Table 3 illustrates the cross validation result for FAM with 3 epochs learning. We describe the circumstances for both KMI and LMI as shown in Fig. 3. Basically, the distance between group *A* and *B* for KMI is smaller ($\delta'_d < \delta_d$) thus creating higher value of vigilance compared to LMI. Meanwhile, the difference between data in the same group for KMI is less than LMI ($\delta'_s < \delta_s$), therefore it generates a slightly smaller number of categories compared to LMI. It seems clear that even though the differences between groups in this case insect object are small, it is always an important factor to investigate the polarity of the data in each group. Therefore the probability for FAM to generate high classification rate is possible.

Table 3: Cross validation result for FAM

		Unit Range			**Improved Unit**			**Improved linear**		
	k	*ρA*	*nA*	*NCC*	*ρA*	*nA*	*NCC*	*ρA*	*nA*	*NCC*
LMI	1	0.96	34	52	0.97	34	52	0.97	37	53
	2	0.96	35	46	0.97	35	46	0.97	34	46
	3	0.96	32	44	0.97	32	44	0.97	33	43
	4	0.96	38	48	0.97	38	48	0.97	38	49
			PCC	79		*PCC*	79		*PCC*	80
KM	1	0.98	30	54	0.98	30	54	0.99	29	56
	2	0.97	34	49	0.98	34	49	0.98	32	50
	3	0.97	24	40	0.98	24	40	0.98	22	41
	4	0.97	33	49	0.98	33	49	0.98	32	50
			PCC	80		*PCC*	80		*PCC*	82

Note: *NCC* = number of correct classification
PCC = percentage of correct classification *nA*= No. of *ARTa* category



a) b)

● = data group *A* ✦ = data group *B*

Fig. 3: A comparison between different dataset a) LMI and b) KMI

5.3 GAM Parameter Initialization

GAM requires only one parameter to be initialized which refers to the gamma parameter *γ*. Other parameters are mathematically obtained from the data pattern that was trained by GAM. Basically, different types of moment generate various data set thus the value of gamma is different for one data set to another. Table 4 illustrates the value that has been tested for different type of moment applied in this work. Note that the value of gamma for KMI is high compared to other moments since KMI produces large numbers for its feature vectors. We found that gamma value is totally different from learning parameters of FAM. The value of gamma principally refers to the standard deviation of data when there is only one input pattern trained for a particular category. Nonetheless, the differences between learning and gamma parameter will be clearly revealed as the discussion progresses. There are four (4) different values of gamma tested for each type of moment applied in the GAM training phase.

We use the highest to the lowest possible value for gamma (from left to right of Table 4). In order to find the optimal value for gamma, Williamson, J.R. [18] suggested to begin the training with a large gamma value decreases its value as training progresses. We found that when the value of gamma is large, the number of epoch training, categories and the vigilance parameter is huge. Moreover, when gamma decreases the value for all those three parameters is also reduced. However when the value of gamma reaches a certain level the number of categories in *ARTa* increases. We manage to determine the vital factor that leads this situation to occur. Fundamentally, this factor has it own unique relation with the use of equation (6) which has been applied to update the standard deviation for every category in *ARTa*. To simplify our discussion, equation (7) is rewritten in an easier form of GAM σ_{ji} parameter update. Fundamentally, when each category learns their first input pattern the value of gamma will be assigned as a standard deviation ($\sigma_{ji} = \gamma$). Nevertheless, when A is too large (γ is too large) compared to B therefore equation (7) becomes (8). These means GAM will take more epochs training to reduce the influence of A in order to produce an optimal value of standard deviation of category *J*. This is shown in Table 4 whereas the number of epoch training is higher when the value of gamma is large.

Table 4: GAM training result using different value of gamma γ

	e	***ρA***	***nA***	***NCC***	***ρA***	***nA***	***NCC***	***ρA***	***nA***	***NCC***
		$\gamma = 0.1$			$\gamma = 0.001$			$\gamma = 0.0001$		
LMI	1	-0.04	21	173	-30	26	240	-179	57	240
	2	-0.005	44	72	-30	26	240	-179	57	240
	3	-0.003	68	152	-30	26	240	-179	57	240
	2	-0.003	93	154	-30	26	240	-179	57	240
	1	-0.003	132	240	-30	26	240	-179	57	240
		$\gamma = 1\text{x}10^{11}$			$\gamma = 1\text{x}10^{9}$			$\gamma = 1\text{x}10^{8}$		
KMI	1	-0.007	27	126	-36	28	240	-83	56	240
	2	-0.001	70	138	-36	28	240	-83	56	240
	3	-0.001	90	135	-36	28	240	-83	56	240
	6	-0.001	142	164	-36	28	240	-83	56	240
	1	-0.001	155	240	-36	28	240	-83	56	240

Note: *NCC* = number of correct classification *e*=epoch *nA*=No. of *ARTa* category

If otherwise *A* is too small compared to *B*, it will make less epoch training for GAM to converge since equation (9) refers to the original standard deviation for category *J*. Using both examples, we can extend our explanation about the differences between large and small gamma in generating the vigilance and categories number in *ARTa*. Fig. 4a illustrates two categories *S* and *N* with large standard deviation in Gaussian distribution space. When σ is large the overlapping area for both categories *S* and *N* become huge. To avoid this area, the value of vigilance parameter in *ARTa* is increased. If there is new input *x* to be fed into GAM, the CMF for the input will be situated below vigilance, thus a new category *Y* will be generated by GAM (Fig. 4b). This is the main reason that there is a large number of categories and vigilance for *ARTa* when gamma is set too large.

$$\sigma_{ji}(t)=\sqrt{(1-n_j^{-1}(t))\sigma_{ji}^2(t-1)+n_j^{-1}(t)(\mu_{ji}(t)-I_i)^2} \qquad (6)$$

$$\sigma_{ji}(t)=\sqrt{\left(1-n_j^{-1}(t)\right)A+n_j^{-1}(t)B} \qquad (7)$$

Where: $A=\sigma_{ji}^2(t-1)$ $\quad B=\left(\mu_{ji}(t)-I_i\right)^2$

$$\sigma_{ji}(t)\cong\sqrt{\left(1-n_j^{-1}(t)\right)A} \qquad (8)$$

$$\sigma_{ji}(t)\cong\sqrt{n_j^{-1}(t)B} \qquad (9)$$

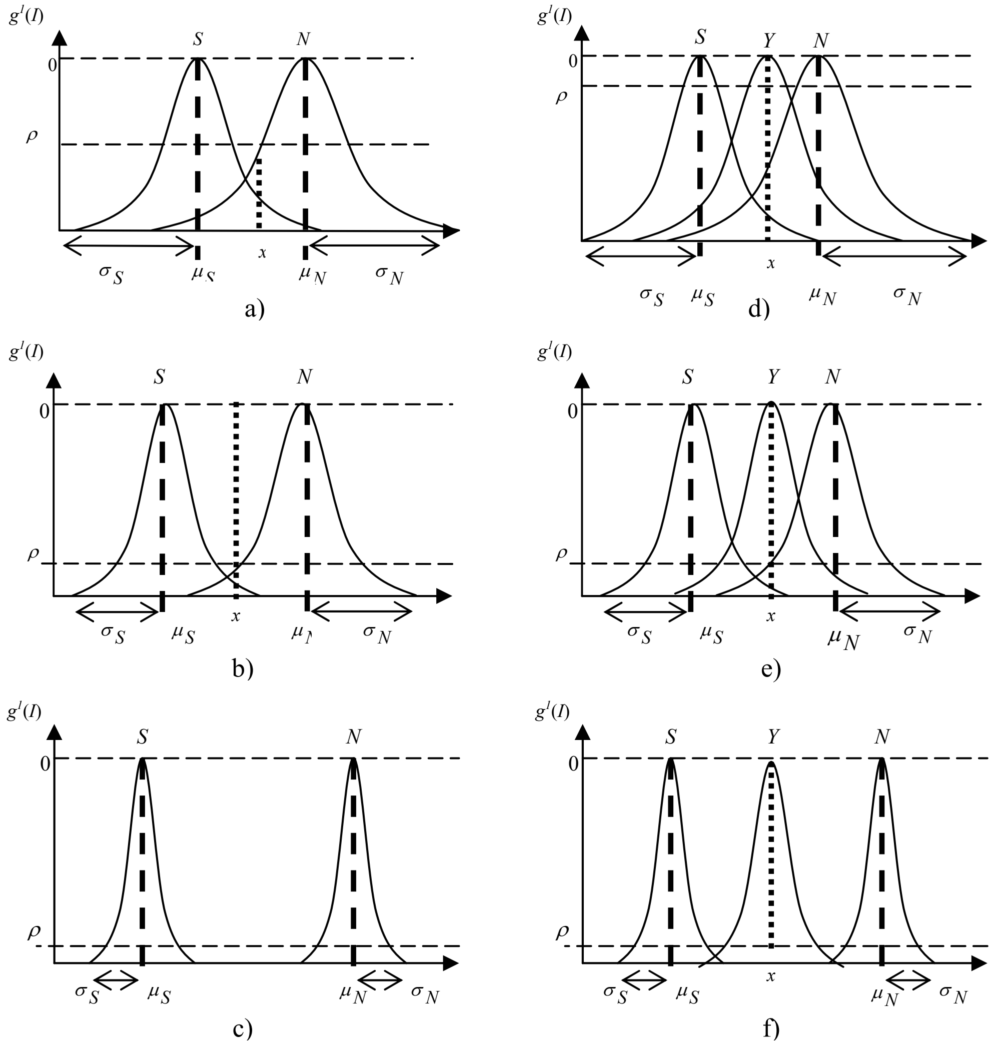


Fig. 4: Gaussian distribution for categories with a) huge γ b) optimal γ c) small γ.
New category crated between two categories with d) huge γ e) optimal γ f) small γ

Meanwhile, when the value of gamma is set to the optimal value, we have the best possible shape for each category as illustrated in Fig. 4c. Therefore, the vigilance is initialized at a considerable value since the overlapping area is small. However, if gamma is set too small we have very narrow shape categories (Fig. 4d). In addition, when the new input x is trained using GAM, thus a new category created (Fig. 4e). This is the main reason for GAM to generate many categories in *ARTa* when gamma value is set too small. The combination of all examples contribute to the fact that certain level of the gamma value causes the situation whereas the number of categories in *ARTa* is not decreasing as the gamma value becomes smaller. Hence, we identify this circumstance as *gamma threshold*. Therefore, we apply gamma value slightly smaller than the *gamma threshold* occurred. For KMI dataset the gamma value used is $1\text{x}10^9$ while for the LMI is 0.001. At this stage we cannot compare specifically the category generated by GAM between different types of moment because the value of gamma applied is basically an approximation only. Further investigation is required to find an accurate value of standard deviation in GAM neural network for every dimension in input feature vector.

5.4 GAM Cross Validation Result

Table 5 illustrates the cross validation result for GAM with 3 epochs learning. Still we can see that GAM produces high classification using KMI dataset. We also found that when the vigilance is too low, the value of *PCC* is small compared to other folds. For example, in ZMI cross validation process, the third fold produces the lowest vigilance if compared to other folds. This situation is also similar in other types of moment data set. Nevertheless, GAM performs better classification of insect images when compared to FAM neural network for all types of moment applied in this work (Table 5).

Table 5: Cross Validation result for GAM

	LMI			***KMI***		
k	*ρA*	*nA*	*NCC*	*ρA*	*nA*	*NCC*
1	-32.44	27	60	-28.61	29	60
2	-23.31	29	59	-28.24	28	60
3	-78.19	20	58	-28.02	21	60
4	-22.13	32	60	-29.22	29	59
		PCC	98.75		*PCC*	99.58

Note: *NCC*=number of correct classification;
PCC=Percentage of correct classification;
k=Number of fold for cross validation;
nA= No. of *ARTa* category

5.5 Comparison between FAM and GAM

From both cross validation result shown in Table 4 and 5, we conclude that GAM demonstrates better performance in classification task when compared to FAM neural network. In addition, we discover three (3) main factors affecting the result for both neural networks in terms of their algorithm characteristics.

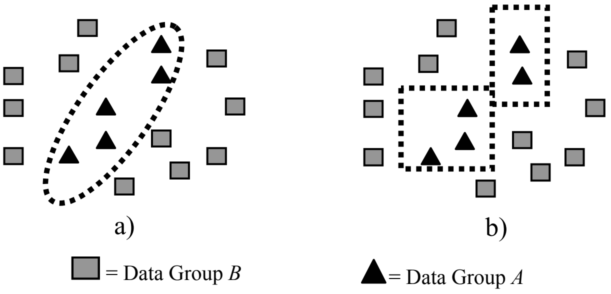


Fig. 5: Clustering technique for a) GAM b) FAM

5.5.1 Different clustering techniques

In principal, any type of neural network based on supervised ARTMAP requires CCF and CMF in order to cluster an input data. Both FAM and GAM mathematically have different CCF and CMF from each other. FAM uses the hyperactangle method to cluster the same input data however GAM utilizes the separable Gaussian distribution. Fig. 5 illustrates an example of input clustering atmosphere. Notice that FAM requires two (2) hyperactangle spaces to cluster the data of group *A* while GAM produces one (1) of its Gaussian distribution space.

5.5.2 Different technique in testing phase

In testing phase, GAM applies different equation [18] compared to FAM while searching the winner category for the given input vector. Since FAM uses vigilance checking in doing so, the chances for FAM will not select any suitable category exist, especially when the current input vigilance is lower compared to *ARTa* vigilance. Meanwhile, GAM applies the probability concept where every input tested will always have their best chosen category even though it is less than the vigilance value.

5.5.3 Normalization Factors

One of the weakness of normalization technique applied is that it causes the lost of vector length information in input vector [2]. This is because the range of the data is only between zero (0) and one (1).

6.0 CONCLUSION

The experimental results demonstrate that GAM offers better classification performance compared to the FAM neural network. This also shows that GAM has an excellent generalization capability while being as artificial classifiers. In fact, the performance of FAM is also influenced by the normalization technique adopted. In this paper we have used three different type of normalization technique that not only generated different classification result but also produced dissimilar number of category in the recognition layer. We have introduced a better technique in order to search the optimum value for gamma γ which is known as gamma threshold value. This will assist us to design the most optimal GAM architecture. We have also demonstrated that GAM can learn the input patterns in finite list of data presentation and proven that GAM can be used in on-line learning situation. However, there is also some weakness within the GAM neural network properties that still need to be investigated further. For example, improvements of GAM category match function (CMF) also can be done since the value of standard deviation for each category are approximation only. In addition, this value of gamma γ will be used either for training or testing in CCF and CMF.

Therefore, the search of winning category to represent the current input pattern seems to be unbalanced between the category that already learned more than one input compared to the category that only learned one input. To make it worse, the fact that every dimension of each features vector should have different value either a small or big number. Therefore, the value of gamma γ can really influence the outcome of GAM performance. Furthermore, we also have demonstrated that *k*-folds cross validation techniques can be used in order to find the generalization capability of supervised based ARTMAP neural network such as GAM and FAM. It also helps to reduce the probability of over-training for both FAM and GAM neural network. It is also found that classification rate for both neural networks are influenced by the type of moment invariant techniques applied. This is proven by KMI which has demonstrated the higher classification rate for FAM and GAM compared to other moment. Finally, we discussed the factors which contribute GAM to achieve better performance than FAM neural network. These factors are clustering techniques, testing methodologies and normalization of input vector.

REFERENCES

[1] G. C. Anagnostopoulos & M. Georgiopulos, “Hypersphere ART and ARTMAP for unsupervised and supervised, incremental learning”, in *Proceedings of the IEEE-INNS-ENNS International Joint Conference on Neural Networks, 2000, IJCNN 2000*, Vol. 6, pp. 59 – 64.

[2] G. C. Anagnostopoulos & M. Georgiopulos, “Ellipsoid ART and ARTMAP for incremental clustering and cla-ssification”, in *International Joint Conference on Neural Networks, 2001, Proceedings IJCNN '01*, Vol. 2, pp. 1221 – 1226.

[3] G. C. Anagnostopoulos et al. (2002). “Reducing generalization error and category proliferation in ellipsoid ARTMAP via tunable misclassification error tolerance: boosted ellipsoid ARTMAP”, in *Proceedings of the 2002 International Joint Conference on Neural Networks, 2002, IJCNN '02*, Vol. 3, pp. 2650 – 2655.

[4] A. Baraldi & E. Alpaydin, “Constructive Feedforward ART Clustering Network - Part I”. *IEEE Transactions on Neural Networks*, Vol. 13 No. 3, 2002, pp. 645-661.

[5] G. A. Carpenter et al., “Fuzzy ARTMAP: A neural network architecture for incremental supervised learning of analog multidimensional maps”, *IEEE Transactions on Neural Networks*, Vol. 3, Issue. 5, 1992, pp. 698 – 713.

[6] G. A. Carpenter et al., (1992). ”Fuzzy ARTMAP: an adaptive resonance architecture for incremental learning of analog maps” in *International Joint Conference on Neural Networks, 1992, IJCNN*, Vol. 3, 1992, pp. 309 – 314.

[7] G. A. Carpenter & S. Grossberg, “A self-organizing neural network for supervised learning, recognition and prediction”. *IEEE Communications Magazine*, Vol. 30, Issue. 9, 1992, pp. 38–49.

[8] M. Georgiopoulos, “Overtraining in fuzzy ARTMAP: Myth or reality?” in *International Joint Conference on Neural Networks, 2001. Proceedings. IJCNN '01*, Vol. 2, pp. 1186 – 1190.

[9] E. Gomez-Sanchez, Y. A. Dimitriadis, J. Cano-Izquierdo & J.M. Lopez-Coronado, "μARTMAP: use of mutual information for category reduction in Fuzzy ARTMAP". *IEEE Transactions on Neural Networks*, Vol. 13, Issue. 1, Jan. 2002. pp. 58 – 69.

[10] P. Henniges, E. Granger & R. Sabourin, "Factors of overtraining with fuzzy ARTMAP neural networks". In *Proceedings of the 2005 IEEE International Joint Conference on Neural Networks, 2005. IJCNN '05,* Vol. 2, pp. 1075 – 1080.

[11] B.W. Jervis, S. Djebali & L. Smaglo, "Integrated probabilistic simplified fuzzy ARTMAP". *IEE Proceedings on Science, Measurement and Technology*, Vol. 151, Issue. 3, 2004, pp. 218 – 228.

[12] M.L.M Lopes, C.R. Minussi & A.D. Lotufo, "Electric load forecasting using a fuzzy ART and ARTMAP neural network". *Applied soft computing*. Vol. 5, Issue. 2, 2004, pp. 235-244.

[13] B. S. Puteh et al., "A Comparison of Feature Normalization Technique on Complex Image recognition" in *The 2nd Conference on Information Technology in Asia (CITA 01)*, 2001.

[14] B. S. Puteh et al., "Effect of Normalization on Rice Yield Prediction" in *National Postgraduate Colloquium 04*, Penang, 8-9 Dec 2004, pp. 330 – 335.

[15] N. Y. Shahrul, *Binary Image Classification using Fuzzy and Gaussian ARTMAP neural networks*. Msc thesis. School of Computer and Communication. KUKUM 2006.

[16] S. J. Verzi et al., "Boosted ARTMAP" in *The 1998 IEEE International Joint Conference on Neural Networks Proceedings, 1998. IEEE World Congress on Computational Intelligence*, Vol. 1, pp. 396 – 401.

[17] S. J. Verzi et al., "Hierarchical ARTMAP" in *Proceedings of the IEEE-INNS-ENNS International Joint Conference on Neural Networks, 2000. IJCNN 2000*, Vol. 6, pp. 41 – 46.

[18] J. R. Wiliamson, "Gaussian ARTMAP: A neural network for fast incremental learning of noisy multi-dimensional maps", *Neural Networks*, Vol. 9, No.5, 1996, pp. 881-897.

BIOGRAPHY

Shahrul Nizam received the B.Eng. in Electrical-Electronic Eng. from University Technology Malaysia in 2004. He obtained his M.Sc. in Computer Engineering from University Malaysia Perlis in 2006. Currently he is working toward Ph.D. degree in Computer system Eng. at University of South Australia. His research areas of interest include the artificial intelligence and machine learning systems. At the moment his is focusing on the agent learning algorithm architecture design for both single and multi-agent systems.

Puteh binti Saad is currently a lecturer at the Department of Software Engineering, Faculty Computer Science and Information Systems, UTM Skudai. She obtained her B.Sc. from University of Malaya, her M.Sc from University of Strathclyde and her Ph.D from UTM. She was attached to School of Computer & Communication Engineering, KUKUM (now known as UNIMAP) temporarily from 2003-2006. Her research interests are on the applications of computational intelligence techniques including bio-inspired computing and swarm intelligence to solve classification and optimization problems.