THE IMPACT OF FEATURE WEIGHTING ON THE FUZZY ARTMAP NEURAL NETWORK

A. CAȚARON[1] L.M.SASU[2] R.ANDONIE[3]

Abstract: *The paper presents an extension of the Fuzzy ARTMAP model that employees the feature weights in the determination of the ART categories. The features weights measure the relative importance of each input in a data classification task.*

Key words: *machine learning, feature weights, fuzzy ARTMAP.*

1. Introduction

The Fuzzy ARTMAP (FAM) architecture is based on the adaptive resonance theory (ART) developed by Carpenter and Grossberg ([3], [4], [5], [7], [8], [10], [14]). FAM family of neural networks is known to be one of the few models that possess incremental learning capability [16], solves the stability-plasticity dilemma [9] and has most of the desirable properties for pattern classifiers [17].

The success of ART-based architectures is given by the advantages they have over another multi-layer networks previously developed [6], [7], [8]: dynamic allocation of nodes without network disruption, fewer training cycles required for training and guaranteed convergence due to using monotonically decreasing weights.

The FAM paradigm is prolific and there are many variations of Carpenter's *et al.* [8] initial model: ART-EMAP [10], dARTMAP [11], Fuzzy ARTVar [13], PROBART [15], Boosted ARTMAP [19], Gaussian ARTMAP [20], FAMR [1], but the list is far to be exhaustive.

FAM with Feature Weighting (FAMFW) [2] algorithm uses a scaled distance measure by considering the input relevance as feature weights. The input relevance represents a numerical quantification of the importance of each feature in a data classification task. In this paper we discuss the details of categories determination while the input weights have different values. In section 2 we give a brief description of FAMFW. Section 3 is dedicated to the presentation of the impact that the feature weighting has in FAMFW. The experiments on artificial data are presented in section 4 while section 5 contains the conclusions.

2. Fuzzy ARTMAP with feature weights

A FAM consists of a pair of fuzzy ART modules, ART_a and ART_b connected by an inter-ART module called Mapfield [8]. ART_a and ART_b are used for coding the input and output patterns, respectively, and Mapfield allows mapping between inputs and outputs. The ART_a module contains the input layer F_1^a and the competitive layer

[1] Dept. of Electronics and Computers, *Transilvania* University of Braşov.
[2] Dept. of Applied Computer Science, *Transilvania* University of Braşov.
[3] Computer Science Department, Central Washington University, Ellensburg, USA.

F_2^a. A preprocessing layer F_0^a is also added before F_1^a. Analogous layers appear in ART_b.

The initial input vectors are in the form:

$$\mathbf{a} = (a_1, \ldots, a_n), a_i \in [0,1] \forall i = 1, \ldots n \tag{1}$$

The *complement coding* is a data preprocessing technique performed in the two fuzzy art module by the F_0^a (and F_0^b respectively) layer in order to avoid proliferation of nodes. Each input vector $\mathbf{a} = (a_1, \ldots, a_n)$ produces the normalized vector $\mathbf{A} = (\mathbf{a}, \mathbf{a}^c) = (\mathbf{a}, 1 - \mathbf{a})$ whose L_1 norm is constant: $|\mathbf{A}| = n$.

Let us denote by M_a the number of nodes in F_1^a and by N_a the number of nodes in F_2^a. Due to the preprocessing step, $M_a = 2n$. $\mathbf{w}^a$ is the weight vector between F_1^a and F_2^a. Each F_2^a node represents a class of inputs grouped together, denoted as a "category". Each F_2^a category has its own set of adaptive weights stored in the form of a vector:

$$\mathbf{w}_j^a = \left(w_{j,1}^a, \ldots, w_{j,M_a}^a\right), \forall j = 1, \ldots N_a \tag{2}$$

where N_a is the number of ART_a categories. They are represented as hyper-rectangles inside the unit box. We use similar notations for the ART_b module that receives m-dimensional input vectors. For a classification problem the class index is the same as the category number in F_2^b, and ART_b can be replaced by an N_b–dimensional vector.

The Mapfield module allows FAM to realize heteroassociations by establishing many-to-one links between various categories from ART_a and ART_a, respectively. The number of nodes in Mapfield is equal to the number of nodes in F_2^b. Each node j from F_2^a is linked to every node from F_2^b via a weight vector $\mathbf{w}_j^{ab}$.

The learning algorithm is shortly described below and a more detailed description can be found in [12]. For every training pattern we set the vigilance parameter factor equal to its baseline value and consider that all nodes are not inhibited. For each preprocessed input $\mathbf{A}$, a fuzzy choice function is used to obtain the response of each F_2^a category:

$$T_j(\mathbf{A}) = \frac{\left|\mathbf{A} \wedge \mathbf{w}_j^a\right|}{\alpha_a + \left|\mathbf{w}_j^a\right|}, \quad j = 1, \ldots N_a \tag{3}$$

Let us denote by J the node with the highest fuzzy choice. If the resonance condition:

$$\rho\left(\mathbf{A}, \mathbf{w}_J^a\right) = \frac{\left|\mathbf{A} \wedge \mathbf{w}_J^a\right|}{|\mathbf{A}|} \geq \rho_a \tag{4}$$

is not fulfilled, then the Jth node is inhibited such that it will not participate to further competitions for this pattern and a new resonant category is searched. This eventually leads to creation of a new category in ART_a.

A similar process occurs in ART_b. If the winning node from ART_b is K, then F_2^b output vector is set to:

$$y_k^b = \begin{cases} 1, \text{ if } k = K \\ 0, \text{ otherwise} \end{cases}, \quad k = 1, \ldots, N_b \tag{5}$$

An output vector $\mathbf{x}^{ab}$ is formed in the Mapfield:

$$\mathbf{x}^{ab} = \mathbf{y}^b \wedge \mathbf{w}_J^{ab} \tag{6}$$

A Mapfield vigilance test controls the match between the predicted vector $\mathbf{x}^{ab}$ and the target vector $\mathbf{y}^b$:

$$\frac{\mathbf{x}^{ab}}{\mathbf{y}^{b}} \geq \rho_{ab} \qquad (7)$$

where $\rho_{ab} \in [0,1]$ is a Mapfield vigilance parameter. If the case when the test from (7) is not passed, then a sequence of steps named *match tracking* is initiated. The vigilance parameter ρ_a is increased and a new resonant category will be sought for ART_a. If the inequality from (7) is fulfilled, then learning occurs in ART_a, ART_b and Mapfield:

$$\mathbf{w}_J^{a(new)} = \beta_a \left(\mathbf{A} \wedge \mathbf{w}_J^{a(old)}\right) + \left(1 - \beta_a\right)\mathbf{w}_J^{a(old)} \qquad (8)$$

and the analogous in ART_b.

In [2] we introduced the FAMFW, an extension of FAM which uses a weighted distance. The size $s(\mathbf{w}_j)$ of a category $\mathbf{w}_j$ is defined by:

$$s(\mathbf{w}_j) = n - \mathbf{w}_j \qquad (9)$$

and the distance between the category $\mathbf{w}_j$ and a normalized input $\mathbf{A}$ is:

$$dis(\mathbf{A}, \mathbf{w}_j) = \left|\mathbf{w}_j\right| - \left|\mathbf{A} \wedge \mathbf{w}_j\right| = \sum_{i=1}^{n} d_{ji} \qquad (10)$$

where $\mathbf{d}_j = \mathbf{w}_j - \mathbf{A} \wedge \mathbf{w}_j = \left(d_{j1}, \ldots d_{jn}\right)$. It was shown in [12] that

$$T_j(\mathbf{A}) = \frac{n - s(\mathbf{w}_j) - dis(\mathbf{A}, \mathbf{w}_j)}{n - s(\mathbf{w}_j) + \alpha_a} \qquad (11)$$

$$\rho(\mathbf{A}, \mathbf{w}_J^a) = \frac{n - s(\mathbf{w}_j) - dis(\mathbf{A}, \mathbf{w}_j)}{n} \qquad (12)$$

The weighted distance $dis(\mathbf{A}, \mathbf{w}_j; \boldsymbol{\lambda})$, a generalized form of $dis(\mathbf{A}, \mathbf{w}_j)$, is:

$$dis(\mathbf{A}, \mathbf{w}_j; \boldsymbol{\lambda}) = \sum_{i=1}^{n} \lambda_i d_{ji} \qquad (13)$$

where $\boldsymbol{\lambda} = (\lambda_1, \ldots \lambda_n)$, $\lambda_i \in [0, n]$ is the weight of the ith feature and $|\lambda| = n$.

3. The impact of weighted distance in FAMFW

First of all, the condition $|\lambda| = n$ allows to set $\lambda_i = 1, \forall i = 1, n$ which leads to the original distance used in FAM. The reason for which we allow zero values for λ_i will be described further in this section. However, other different restrictions on λ could be imagined.

Below we present some remarks concerning the properties of weighted distance used in FAMFW, considered for the bidimensional input space.

When we set $\lambda_i = 1, \forall i = 1,2$ the points situated at a constant distance from the input category corresponds to a hexagonal shape, as shown in Figure 1. When one considers that a specific feature has a different weight from the other, the hexagonal shape is flattened on the direction of the feature with the highest weight. This is depicted in Figure 2, when the feature along the *y* axis has a larger weight that the one from the *x* dimension.

This behavior is in accordance with our intuition: for features that are found to be more relevant than the others, we should avoid large shapes on that direction, as these could lead to overlaps and misclassification, or worse, to rejected training patterns. Meanwhile, for the dimensions where we have low discriminatory power, denoted by low values of λ_i, one should allow for large shapes, as these will not lead to overlaps. This intuition is supported by the experimental results.

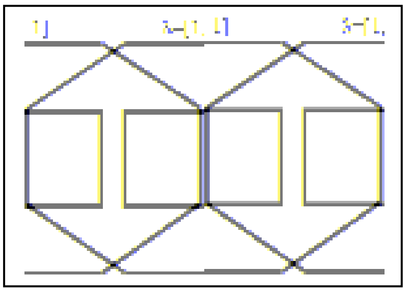

Figure 1. *Points located at constant distance from the input category*

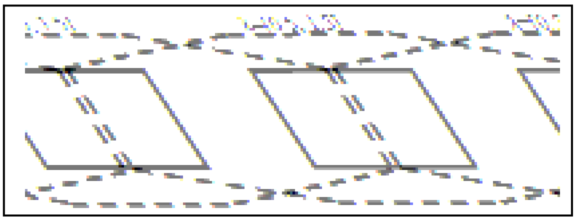

Figure 2. *Points located at constant distance from the category, when the weight associated to the vertical axis is larger than the one for the x axis*

It can be easily shown that when $\lambda_i > 0, \forall i = 1,2$, the shapes generated by points at constant weighted distance from the category are always hexagonal. When one of the two weights is null, the shape reduces to a pair of straight lines inside the unit square, situated on one side and another of the rectangle representing the category. This is shown in Figures 3 and 4. Again, this is fully compliant with the general intuition, as irrelevant features will produce shapes that has no variance along the corresponding dimension.

4. Experimental results

We illustrate the FAMFW behavior on two artificial datasets, consisting of points selected from the surfaces determined by two circles. The main purpose of this study is to show how the feature weights influence the behavior and the performance of FAMFW.

We generated two datasets. The first one represents bidimensional patterns consisting of points coordinates randomly generated inside of two overlapping circles, and the second one contains patterns from two non-overlapping circles. In both cases the *x* coordinates of the two circles have the same values (see Figures 5 and 6) and each of them is considered an output class. The points outside the circles were not taken into account.

The training set consisted of 20 patterns and the test set contained 10000 patterns, with approximately the same number of patterns for each class. Each time we performed five tests with different pairs of train/test sets. The points are uniformly, independently and identically distributed inside the circles. For the overlapping circles, the centers were (50, 50) and (50, 100), respectively, and for the non-overlapping circles (50, 50) and (50, 170). In each case the radius was 50 points.

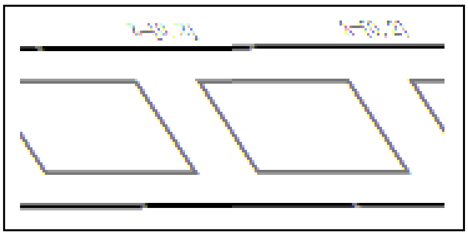

Figure 3. *Point situated at constant distance from the rectangle, when the feature along the x axis has null weight*

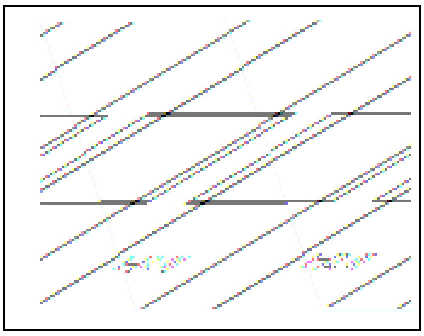

Figure 4. *Point situated at constant distance from the rectangle, when the feature along the y axis has null weight*

We used the overlapped circles in tests with the following values of the feature weights: $\lambda = [1,1]$ which is the particular case of FAM, $\lambda = [0.8,1.2]$, $\lambda = [0.5,1.5]$, and $\lambda = [0,2]$. Because we have been particularly interested by the overlapping classes, we only reported here for the case of non-overlapped circles the tests realized with $\lambda = [1,1]$ and $\lambda = [0,2]$. The individual and averaged values for the number of ART_a categories and the percent of correct classification (PCC) are reported in Tables 1 and 2 for the overlapping and the non-overlapping circles, respectively. For the last three values of λ, the number of ART_a categories is relatively the same as for FAM. However, the PCC increases with the weight associated to the *y* feature, reaching the highest value when the *x* feature is completely omitted, i.e. when $\lambda = [0,2]$. The PCC, for increasing value of the second weight, is monotonically increasing and at least as large as the PCC obtained for $\lambda = [1,1]$.

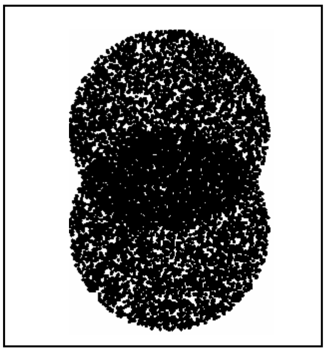

Figure 5. *Test set for two overlapping circles centered at (50,50) and (50,100), with the radius of 50 points*

We also tested the behavior of FAMFW when the weights are given contrary to the general intuition, *i.e.* large for non-relevant features and small for relevant features. The results are given in Tables 3, for the non-overlapping circles. As one can see, the performance degrades significantly for these cases, in terms of PCC. Similar behavior was obtained for the case of overlapping circles.

Table 1

Number of ART_a categories and PCC for overlapping circles and various sets of λ.

Test no	$\lambda=[1,1]$		$\lambda=[0.8,1.2]$		$\lambda=[0.5,1.5]$		$\lambda=[0,2]$	
	No. of ART_a categ	**PCC**	**No. of ART_a categ**	**PCC**	**No. of ART_a categ**	**PCC**	**No. of ART_a categ**	**PCC**
1	9	70.59%	9	71.84%	10	72%	9	72.16%
2	8	68.73%	8	69.17%	9	68.58%	8	68.87%
3	8	72.22%	7	73.37%	8	74.99%	8	77.64%
4	8	76.31%	9	76.96%	8	78.99%	11	77.82%
5	8	78.56%	9	78.82%	9	78.31%	8	77.59%
Avg.	8.2	73.282%	8.4	74.032%	8.8	74.574%	8.8	74.816%

Table 2

Number of ART_a categories and PCC for non-overlapping circles and various sets of λ.

Test no	$\lambda=[1,1]$		$\lambda=[0,2]$	
	No. of ART_a categories	**PCC**	**No. of ART_a categories**	**PCC**
1	6	98.14%	7	99.95%
2	6	99.25%	7	99.7%
3	6	98.92%	5	100%
4	6	100%	8	100%
5	6	99.21%	5	100%
Avg.	6	99.104%	6.4	99.93%

Table 3

The effect of using inappropriate values for weighting the features (PCC and ART_a categories number) in the case of the two non-overlapping circles. These weights reverse the natural importance of the two features

Test no	$\lambda=[1.2,0.8]$		$\lambda=[1.5,0.5]$		$\lambda=[2,0]$	
	No. of ART_a categories	**PCC**	**No. of ART_a categories**	**PCC**	**No. of ART_a categories**	**PCC**
1	6	97.13%	7	98.15%	6	49.79%
2	6	98.33%	6	91.97%	7	50.44%
3	6	98.61%	5	99.05%	7	49.39%
4	7	100%	7	99.46%	8	49.42%
5	6	98.99%	5	98.6%	7	50.5%
Avg.	6.2	98.612%	6	97.446%	7	49.908%

4. Conclusions

The feature weights introduced in Fuzzy ARTMAP determine a more "sensitive" behaviour on the dimensions with a higher importance for the patterns classification, while the regions on the less important dimensions can more easily expand. The tests with the artificial dataset had one completely irrelevant feature and we progressively decreased its weight. The first goal of our tests was to determine the recognition accuracy of the FAMFW with different weights values. The second goal of our tests was to determine the behaviour of FAMFW when we artificially reverse the importance of the two features by assigning to them inappropriate weights.

In the case of the overlapping circles, the progressive increase of the importance of the second feature produced a better recognition accuracy for the FAMFW. On the other hand, as we expected, the number of ART_a categories slightly increased because the shape of the regions was influenced by the values of the weights. The datasets obtained from the non-overlapping circles was, as we expected, an easier task for FAMFW and the zero value of the least important feature produced the best recognition accuracy.

For the second set of tests we assigned to the most important feature a lower weights that to the other feature. The feature weights were reversed and we tried to verify that FAMFW will have lower performance. The PCC values decreased comparing to the value obtained for $\lambda = [0,2]$. The results confirmed our suppose and the weights vector $\lambda = [2,0]$ produced a PCC of 49.908%, very close to 50% which is the highest non-determination.

Our experiments showed that FAMFW is a viable model. An open problem that might be investigated in the future is what is the limit of the overlapping degree that the classes can have.

References

1. Andonie, R., Sasu, L., Beiu, V.: *Fuzzy ARTMAP with Relevance Factor.* In: Proceedings of the International Joint Conference on Neural Networks (IJCNN 2003), Portland, Oregon, July 20-24 2003, p. 1975-1980.
2. Andonie, R., Cațaron, A., Sasu, L.M.: *Fuzzy ARTMAP with Feature Weighting.* In: Artificial Intelligence and Applications AIA 2008, Innsbruck, Austria, February 13-15, 2008, accepted.
3. Carpenter, G. A., Grossberg, S.: *A massively parallel architecture for a self-organizing neural pattern recognition machine.* In: Computer Vision, Graphics and Image Processing, vol. 37, 1987, p. 54-115.
4. Carpenter, G. A., Grossberg, S.: *ART 2: Self organization of stable category recognition codes for analog input patterns.* In: Applied Optics, vol. 26, 1987, p. 4919-4930.
5. Carpenter, G. A., Grossberg, S.: *ART 3: Hierarchical search using chemical transmitters in self-organizing patterns recognition architectures.* In: Neural Networks, vol. 3, 1989, p. 129-152.
6. Carpenter, G. A., Grossberg, S., Rosen, D. B.: *Fuzzy ART: fast stable learning and categorization of analog patterns' by an adaptive resonance system.* In: Neural Networks, vol. 4, 1991, p. 759-771.
7. Carpenter, G. A., Grossberg, S., Reynolds, J. H.: *ARTMAP: Supervised real-time learning and classification of nonstationary data by a self-organizing neural network.* In: Neural Networks, vol. 4, 1991, p. 565-588.

8. Carpenter, G. A., Grossberg, S., Markuzon, N., Reynolds, J. H., Rosen, D. B.: *Fuzzy ARTMAP: A Neural Network Architecture for Incremental Supervised Learning of Analog Multidimensional Maps.* In: IEEE Transactions on Neural Networks, vol. 3, no. 5, 1992, p. 698-713.
9. Carpenter, G. A., Grossberg, S.: *The ART of Adaptive Pattern Recognition by a Self-Organizing Neural Network.* In: IEEE Computer, vol. 21, no. 3, 1998, p. 77-88.
10. Carpenter, G. A., Ross, W.: *ART-EMAP: A neural network architecture for learning and prediction by evidence accumulation.* In: IEEE Transactions on Neural Networks, vol. 6, no. 4, 1995, p. 805-818.
11. Carpenter, G. A., Milenova, B. L., Noeske, B. W.: *Distributed ARTMAP: A neural network for fast distributed supervised learning.* In: Neural Networks, vol. 11, no. 4, 1998, p. 793-813.
12. Charalampidis, D., Anagnostopoulos, G., Georgiopoulos, M., Kasparis, T.: *Fuzzy ART and Fuzzy ARTMAP with Adaptively Weighted Distances.* In: Proceedings of the SPIE, Applications and Science of Computational Intelligence, Aerosense 2002, Vol. 4739, p. 86-97.
13. Dagher, I., Georgiopoulos, M., Heileman, G. L., Bebis, G.: *Fuzzy ARTVar: An Improved Fuzzy ARTMAP Algorithm.* In: Proceedings of the IEEE World Congress on Computational Intelligence WCCI'98, 1998, p. 1688-1693.
14. Grossberg, S.: *How does a brain build a cognitive code?* In: Psychological Review, vol. 1, 1980, p. 1-51.
15. Marriott, S., Harrison, R. F.: *A Modified Fuzzy ARTMAP Architecture for the Approximation of Noisy Mappings.* In: Neural Networks, vol. 4, no. 8, 1995, p. 619-641.
16. Polikar, R., Udpa, L., Udpa, S. S., Honovar, V.: *Learn++: An incremental learning algorithm for supervised neural networks.* In: IEEE Transactions on Systems, Man, and Cybernetics--Part C, vol. 31, no. 4, 2001, p. 497-508.
17. Simpson, P. K.: *Fuzzy Min--Max Neural Networks - Part 1: Classification.* In: IEEE Transactions on Neural Networks, vol. 3, 1992, p. 776-786.
18. Taghi, M., Baghmisheh, V., Pavesic, N.: *A Fast Simplified Fuzzy ARTMAP Network.* In: Neural Processing Letters, vol. 17, no. 3, 2003, p. 273-316.
19. Verzi, S. J., Heileman, G. L., Georgiopoulos, M., Healy, M. J.: *Boosted ARTMAP.* In: Proceedings of the IEEE World Congress on Computational Intelligence WCCI'98, 1998, p. 396-400.
20. Williamson, J.: *Gaussian ARTMAP: A neural network for fast incremental learning of noisy multidimensional maps.* In: Neural Networks, vol. 9, no. 5, 1996, p. 881-897.

Impactul ponderării intrărilor în rețelele neurale Fuzzy ARTMAP

Rezumat: *Lucrarea prezintă o extensie a modelului Fuzzy ARTMAP care folosește ponderarea intrărilor pentru determinarea categoriilor ART. Ponderile intrărilor cuantifică importanța relativă a fiecărei intrări în clasificarea datelor*

Cuvinte cheie: *machine learning, ponderea intrărilor, fuzzy ARTMAP.*
Recenzent:.Prof. dr. ing. Gheorghe Toacşe
Supervizor traducere în limba engleză: Asist. univ. Laura Sasu