Fuzzy ART and Fuzzy ARTMAP with Adaptively Weighted Distances

Dimitrios Charalampidis[*] [a], Georgios C. Anagnostopoulos** [b], Michael Georgiopoulos*** [b], Takis Kasparis[****] [b]

[a]EE Dept., University of New Orleans
[b]School of EE & CS, University of Central Florida

ABSTRACT

In this paper, we introduce a modification of the Fuzzy ARTMAP (FAM) neural network, namely, the Fuzzy ARTMAP with adaptively weighted distances (FAMawd) neural network. In FAMawd we substitute the regular L_1-norm with a weighted L_1-norm to measure the distances between categories and input patterns. The distance-related weights are a function of a category's shape and allow for bias in the direction of a category's expansion during learning. Moreover, the modification to the distance measurement is proposed in order to study the capability of FAMawd in achieving more compact knowledge representation than FAM, while simultaneously maintaining good classification performance. For a special parameter setting FAMawd simplifies to the original FAM, thus, making FAMawd a generalization of the FAM architecture. We also present an experimental comparison between FAMawd and FAM on two benchmark classification problems in terms of generalization performance and utilization of categories. Our obtained results illustrate FAMawd's potential to exhibit low memory utilization, while maintaining classification performance comparable to FAM.

Keywords: adaptive resonance theory, Fuzzy ART, Fuzzy ARTMAP, classification

1. INTRODUCTION

FuzzyARTMAP[1] (FAM) is a neural network architecture with deep roots in the *adaptive resonance theory* (ART) paradigm[2] proposed by Grossberg. The primary purpose of FAM is to learn cluster associations (mappings) between an input and an output domain. As a special case, it is capable of performing classification tasks, when the output domain coincides with a set of pertinent class labels. FAM is based of *Fuzzy ART*[3] (FA), which is used for clustering. To perform its learning task, FAM (and FA) summarizes similar input patterns of the same class label into *FA categories*. These categories are constructed in a self-organizing fashion and constitute the building block of knowledge/memory representation. Furthermore, their geometric representations are axis-parallel hyper-rectangles embedded in the data domain. It is the use of categories for clustering that makes FAM an *exemplar-based* architecture. FAM features several desirable characteristics. First, it is simultaneously capable of off-line (batch) and on-line (incremental) learning. In the first mode FAM learns its task after repeatedly presenting the entire training set, while in the second mode FAM incrementally incorporates new evidence as it becomes available. Under fast learning rule[1] assumptions, FAM exhibits fast, stable and finite off-line learning: the networks' knowledge stabilizes relatively fast after a finite number of *list presentations* (epochs). Furthermore, due to the nature of its architecture, it is easy to explain FAM's responses to specific input patterns, in contrast to other neural network models, where in general it is difficult to explain why an input pattern **x** produced an output **y**. An important characteristic of FAM is the employment of novelty detection mechanisms that identify input patterns not typical in relation to previously experienced inputs. This novelty detection capability of FAM constitutes the backbone of its *match-based* learning.

[*] dcharala@uno.edu; Phone: 1 504 280-7415; 1224 Lake Avenue #313, Metairie, LA USA 70005; [**] anagnostop@email.com; phone 1 407 228-1231; http://www.geocities.com/g_anagnostop; 4132 Lake Underhill Road #303, Orlando, FL USA 32803; [***] michaelg@mail.ucf.edu; phone 1 407 823-5338; fax 1 407 823-5835; School of EECS, University of Central Florida, 4000 Central Florida Blvd., Orlando, FL USA 32816-2450; kasparis@pegasus.cc.ucf.edu; Phone: 1 407 823-5913; fax 1 407 823-5835; School of EECS, University of Central Florida, 4000 Central Florida Blvd., Orlando, FL USA 32816-2450.

Applications and Science of Computational Intelligence V, Kevin L. Priddy, Paul E. Keller, Peter J. Angeline, Editors, Proceedings of SPIE Vol. 4739 (2002) © 2002 SPIE · 0277-786X/02/$15.00

FAM's representation and learning rules induce a L_1-norm geometry. For example, distances of input patterns from clusters are measured using the L_1 (city-block, Manhattan) distance. Our intent in this paper is to study a modification of the basic FAM architecture, which we call *Fuzzy ARTMAP with adaptively weighted distances* (FAMawd). In the introduced architecture distances are measured using a weighted version of the L_1 distance, whose weights depend on the shape of the category (cluster). This allows for increased selectivity of direction in the input domain, when categories expand during learning. The modification to the distance measurement is proposed to study the capability of FAMawd to achieve more compact knowledge representation than FAM, while simultaneously maintaining good classification performance. Towards that goal, we compare the original FAM to FAMawd on two benchmark classification problems in terms of both performance and number of categories employed. The obtained results approve to be very favorable for the case of FAMawd, which is capable of achieving comparable generalization to FAM utilizing, though, fewer categories.

The rest of the paper is organized as follows. In Section 2 we provide some background on FAM, such as the description of FA categories and the architecture's two phases of operation. In Section 3 we introduce the modification to FAM that gives rise to FAMawd. Section 4 presents the results of our simulations and, finally, in Section 5 we summarize our approach and experimental findings.

2. ELEMENTS OF FUZZY ARTMAP

A block diagram of FAM is depicted in Figure 1. The architecture consists of two FA modules, ART_a and ART_b, that cluster patterns from the input and output space respectively. These two modules are interconnected with an *inter-ART* module (*map field*), which stores the information regarding the associations between input and output clusters. It is assumed that the data domain for each FA module is the set U^M (unit hyper-cube), where U=[0,1] and M is the dimensionality of the domain. Moreover, each FA module consists of three layers: F_o, F_1 and F_2. Layer F_o performs *complement coding* on the input patterns. Complement coding refers to the transform of an input row vector $\mathbf{x}$ to $\mathbf{x}^e=[\mathbf{x}\ \mathbf{x}^c]=[\mathbf{x}\ \mathbf{1}\text{-}\mathbf{x}]$, where $\mathbf{1}$ is the all-ones row vector. The complement-coded patterns $\mathbf{x}^e$ are eventually presented to the F_1 layer (*presentation layer*). Layers F_1 and F_2 are interconnected via top-down and bottom-up connections and each one of these connections feature a weight. In specific, the vector $\mathbf{w}_j$ consisting of all top-down weights emanating from F_2-layer node j is called *template* of j. The F_2 layer (*representation layer*) itself is a MAXNET featuring inhibitory lateral connections and consists of two types of nodes: *committed* and *uncommitted.* Committed nodes feature a template that contains the description of a single *FA category* (cluster) that has been learned via training and which summarizes the patterns that have been encoded by the corresponding F_2-layer node. In contrast, uncommitted nodes do not correspond to real categories and constitute the blank memory of the system. Finally, each FA module also contains a *reset node* to disable F_2-layer nodes under certain circumstances, which are going to be clarified later in the text. As we have mentioned earlier, cluster associations are learned by the map field of FAM. During training, when ART_a chooses to incorporate a specific pattern into a category, such that it would violate an already learned association to a second category in ART_b, FAM responds by performing a *lateral reset* of the offending F_2-layer node in ART_a and engages *match tracking* (MT). Let us note that for classification tasks the role of the ART_b module in FAM becomes trivial. The output domain is the set of class labels pertinent to the classification problem at hand and ART_b will create a distinct FA category for each class label. Since FAM finds more applications as a classifier, in the future, when we will refer to FAM and its network parameters, we will mean the FAM classifier and its ART_a module's network parameters. Also, when we will refer to F_2-layer nodes we will refer to the ones in FAM's ART_a module.

The geometric representation of an FA category (or simply, category) is a hyper-rectangle embedded in the input domain of FAM's ART_a module. It is the description of this hyper-rectangle that is stored in the category's template. For a F_2-layer node j its template $\mathbf{w}_j$ is of the form $\mathbf{w}_j=[\mathbf{u}_j\ \mathbf{v}_j^c]\in U^{2M}$, where $\mathbf{u}_j$, $\mathbf{v}_j\in U^M$ are its *template elements*. Due to FAM's learning scheme, for every template it always holds that $u_{jm}\leq v_{jm}$ with m=1..M. Therefore, the template elements uniquely define a category, whose geometric representation is a hyper-rectangle. Uncommitted nodes feature a template of $\mathbf{w}_u=w_u\mathbf{1}$ with $\mathbf{w}_u\in U^{2M}$, where $w_u\geq 1$ is one FAM's network parameters. Let $|\cdot|:U^{2M}\rightarrow R$ to be the L_1 norm for the U^{2M} domain and $\|\cdot\|_1:U^M\rightarrow R$ to be the L_1 norm for the U^M domain. Additionally, let us define the *min-operator* $\wedge:U^{2M}\times U^{2M}\rightarrow U^{2M}$, such that, if $\mathbf{w}_1$, $\mathbf{w}_2\in U^{2M}$ and $\mathbf{w}_3=\mathbf{w}_1\wedge\mathbf{w}_2$, then the m^{th} component w_{3m} of vector $\mathbf{w}_3$ is $w_{3m}=\min\{w_{1m}, w_{2m}\}$.

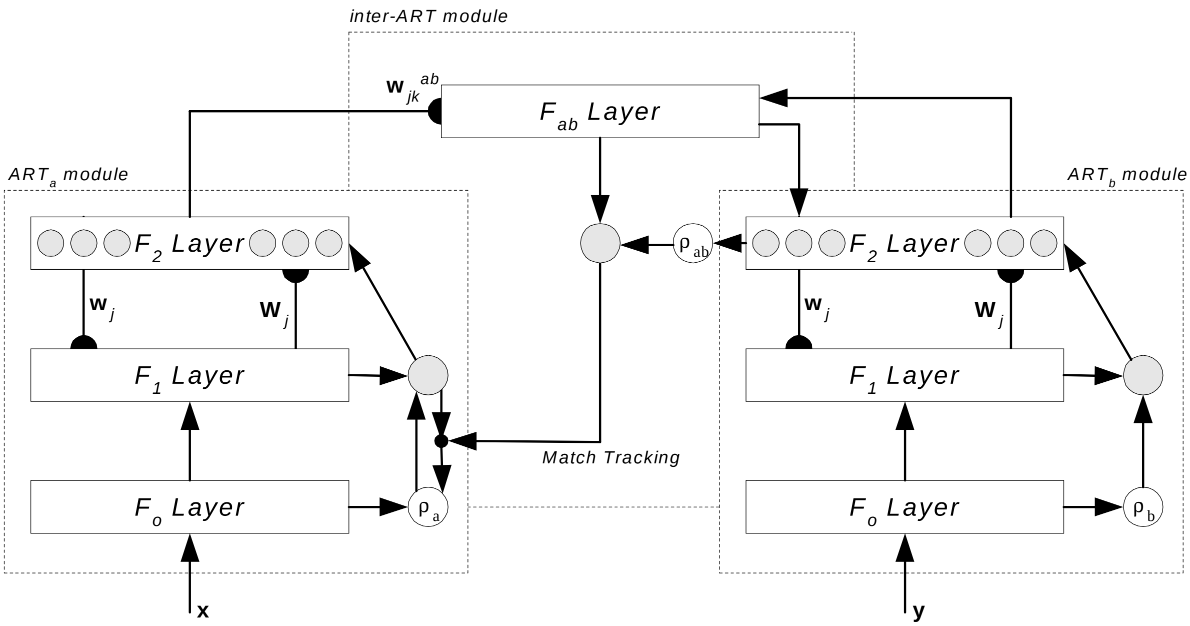

Figure 1: Block diagram of Fuzzy ARTMAP.

Then we define as the *size* $s(\mathbf{w}_j)$ of a category j the quantity

$$s(\mathbf{w}_j) = M - \left|\mathbf{w}_j\right| = \left\|\mathbf{v}_j - \mathbf{u}_j\right\|_1 = \sum_{m=1}^{M}\left(v_{jm} - u_{jm}\right). \tag{1}$$

Note, that for every input pattern $\mathbf{x} \in U^M$ and template $\mathbf{w}_j$ it holds

$$\begin{aligned} \left|\mathbf{w}_j\right| &= M - \left\|\mathbf{v}_j - \mathbf{u}_j\right\|_1 = M - \sum_{m=1}^{M}\left(v_{jm} - u_{jm}\right) \\ \left|\mathbf{x}^e \wedge \mathbf{w}_j\right| &= 2M - \sum_{m=1}^{M}\left[\max\left\{x_m, v_{mj}\right\} - \min\left\{x_m, u_{mj}\right\}\right] \end{aligned}. \tag{2}$$

Based on Equations 1 and 2, we define as the *distance* of a pattern $\mathbf{x} \in U^M$ from category j the quantity

$$dis(\mathbf{x}, \mathbf{w}_j) = \left|\mathbf{w}_j\right| - \left|\mathbf{x}^e \wedge \mathbf{w}_j\right| = \sum_{m=1}^{M}\left[\left(\max\left\{x_m, v_{jm}\right\} - v_{jm}\right) + \left(u_{jm} - \min\left\{x_m, u_{jm}\right\}\right)\right]. \tag{3}$$

Also, we define as *representation region* $R(\mathbf{w}_j)$ of a category (committed node) j with template $\mathbf{w}_j$ the following subset of U^M

$$R(\mathbf{w}_j) = \left\{\mathbf{x} \in U^M \mid dis(\mathbf{x}, \mathbf{w}_j) = 0\right\} \overset{Eq.3}{\Rightarrow} R(\mathbf{w}_j) = \left\{\mathbf{x} \in U^M \mid \mathbf{x}^e \wedge \mathbf{w}_j = \mathbf{w}_j\right\}. \tag{4}$$

An example of a FA category j for 2-dimensional input space is depicted in Figure 2. The union of the illustrated shaded area and the boundaries of the rectangle defined by $\mathbf{u}_j$ and $\mathbf{v}_j$, corresponds to the representation region $R(\mathbf{w}_j)$ of category j. Also depicted in the same figure, $dis(\mathbf{x},\mathbf{w}_j)$ reflects the minimum L_1 distance (also known as *city-block* or *Manhattan* distance) between pattern $\mathbf{x}$ and category's j representation region. Note that, if $\mathbf{x}$ were inside or on the borders of the rectangle, its distance from category j would have been 0.

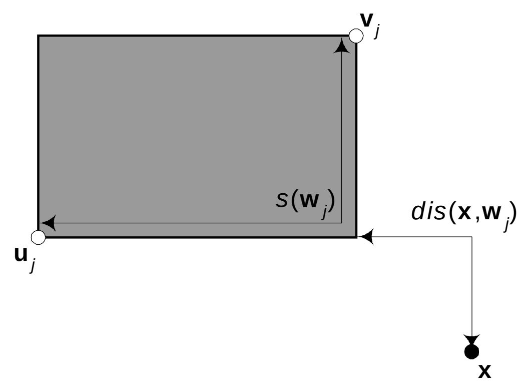

Figure 2: Geometric representation of FA category *j* assuming a 2-dimensional input space.

FAM has two modes of operation: *training phase* and *performance phase*. Both phases are of similar nature except that during performance phase no learning is performed. Learning itself in FAM is achieved by creating and then updating categories. More specifically, during its training phase FAM incrementally clusters input domain data into categories by committing F_2-layer nodes and by updating appropriately their templates. Simultaneously it forms associations of categories to class labels via its map field. Before any learning takes place, all F_2-layer nodes are uncommitted. As more knowledge is accumulated about the pattern domain during training, F_2-layer nodes become gradually committed.

Upon presentation of a specific input pattern $\mathbf{x}$, F_2-layer nodes compete in terms of their *category choice function* (CCF – also known as *activation function*) values. In this paper we consider two types of CCF, namely the *Weber Law* (WL) and the *Choice-by-Difference*[4] (CBD), which are given in Equations 5 and 6 respectively.

$$T(\mathbf{w}_j \mid \mathbf{x}) = \frac{\left|\mathbf{x}^e \wedge \mathbf{w}_j\right|}{\left|\mathbf{w}_j\right| + a} \overset{Eq.1,3}{=} \frac{M - s(\mathbf{w}_j) - dis(\mathbf{x},\mathbf{w}_j)}{M - s(\mathbf{w}_j) + a}. \tag{5}$$

$$T(\mathbf{w}_j \mid \mathbf{x}) = (1-a)\left(2M - \left|\mathbf{w}_j\right|\right) + \left|\mathbf{x}^e \wedge \mathbf{w}_j\right| \overset{Eq.1,3}{=} M(2-a) - as(\mathbf{w}_j) - dis(\mathbf{x},\mathbf{w}_j). \tag{6}$$

In the above definitions a is a FAM network parameter; for the case of the WL choice function $a \in (0,\infty)$ and for the case of CBD choice function $a \in (0,1)$. During the competition the node of the highest CCF value $T(\mathbf{w}_j|\mathbf{x})$ and smallest index j is declared as being the winner. Uncommitted nodes also participate in the competition with a constant CCF value of T_u for all input patterns $\mathbf{x}$ that is parameterized by w_u. The expressions of T_u for the WL and CBD choice functions are given in Equations 7 and 8 respectively.

$$T_u = \frac{M}{2Mw_u + a}. \tag{7}$$

$$T_u = M\left[1 - 2(1-a)(w_u - 1)\right]. \tag{8}$$

The competition of a committed node j against an uncommitted is termed as *commitment test* (CT). Category j satisfies the CT for a given pattern **x** if the following inequality holds

$$T(\mathbf{w}_j \mid \mathbf{x}) \geq T_u \, . \qquad (9)$$

Non-satisfaction of Equation 9 means that **x** will choose an uncommitted node over node j. The CT is implicitly performed during the node competition for pattern selection that we have described here.

Upon completion of identifying the winning node J (which might be committed or uncommitted), its *category match function* (CMF) value $\rho(\mathbf{w}_J|\mathbf{x})$ is being calculated. The CMF for a committed node j is given in Equation 10, while for an uncommitted node j it is defined to be constant $\rho(\mathbf{w}_j|\mathbf{x})=1$ for all patterns **x**.

$$\rho(\mathbf{w}_j \mid \mathbf{x}) = \frac{\left|\mathbf{x}^e \wedge \mathbf{w}_j\right|}{M} \overset{Eq.1,3}{=} \frac{M - s(\mathbf{w}_j) - dis(\mathbf{x}, \mathbf{w}_j)}{M} \, . \qquad (10)$$

Next, the CMF value $\rho(\mathbf{w}_J|\mathbf{x})$ of J is compared to the vigilance network parameter value $\rho \in [0,1]$. This comparison, which is called *vigilance test* (VT), checks if the following inequality holds:

$$\rho(\mathbf{w}_J \mid \mathbf{x}) \geq \rho \, . \qquad (11)$$

If Equation 11 is not satisfied, then J is temporarily reset via the module's reset node (*mismatch reset* – its CCF value is set to zero) until the presentation of the next input pattern. With J effectively excluded from the competition, the search continues for the category featuring the highest CCF value among the remaining ones. If all categories become gradually reset, an uncommitted node will eventually emerge as the winning node; then we say that **x** chooses an uncommitted node. On the other hand, if Equation 11 is true, we say that pattern **x** chooses node (category) J.

If FAM is operating in its performance phase, the network reports the class label associated to category J as being the class label of **x**. On the other hand, if FAM is operating in training phase, a *prediction test* (PT) is performed: if the class label associated to J differs from the class label of pattern **x**, letting J be modified by **x** would cause its misclassification. Therefore, a lateral reset will occur, so that $T(\mathbf{w}_J|\mathbf{x})$ is reset to 0 and node J will be effectively removed from the node competition process until the next input pattern is presented. Next, MT is engaged, which temporarily raises the vigilance ρ from its baseline value to a new value of $\rho(\mathbf{w}_J|\mathbf{x})+\varepsilon$ with $\varepsilon<<1$ and the node competition process resumes until finally a suitable node has been identified that was chosen by **x** and that makes the correct class label prediction. Then, the vigilance parameter ρ is set back to its baseline value. At this point let's assume that the class label associated to J coincides with the class label of pattern **x**. The final step in the learning process is to modify J's template in light of the new information **x**. If J is an uncommitted node, then **x** will initiate the creation of a new category by modifying its template to

$$\mathbf{w}_J^{new} = \mathbf{x}^e \, . \qquad (12)$$

This type of category and any other category j that features $s(\mathbf{w}_j)=0$ is called a *point category*. If J is an already committed node with template $\mathbf{w}_j^{old}$, its template will be updated to $\mathbf{w}_j^{new}$ as follows:

$$\mathbf{w}_J^{new} = \gamma\left(\mathbf{x} \wedge \mathbf{w}_J^{old}\right) + (1-\gamma)\mathbf{w}_J^{old} \, . \qquad (13)$$

where $\gamma \in (0,1]$ is the *learning rate* parameter of FAM. When $\gamma=1$, FAM is said to be performing *fast learning*, otherwise *slow learning*. The learning rule in Equation 13 causes the representation region of J to expand in the direction of **x**. In particular, for fast learning its new representation region will be the minimum hyper-volume hyper-rectangle that contains both the old representation region and pattern **x**. For any value of the learning rate it can be shown from Equations 1,2,3 and 13 that the size of J and the distance of **x** from J after the update are given by

$$s(\mathbf{w}_J^{new}) = s(\mathbf{w}_J^{new}) + \gamma \, dis(\mathbf{x}, \mathbf{w}_J^{old}) \, . \qquad (14)$$

$$dis(\mathbf{x}, \mathbf{w}_J^{new}) = (1-\gamma) dis(\mathbf{x}, \mathbf{w}_J^{old}) . \tag{15}$$

Figure 3 illustrates the above observations for an update of a 2-dimensional category *j*. Updated template elements $\mathbf{u}_j^{new1}$, $\mathbf{v}_j^{new1}$ result, when slow learning is in effect, and elements $\mathbf{u}_j^{new2}$, $\mathbf{v}_j^{new2}$, when fast learning is used.

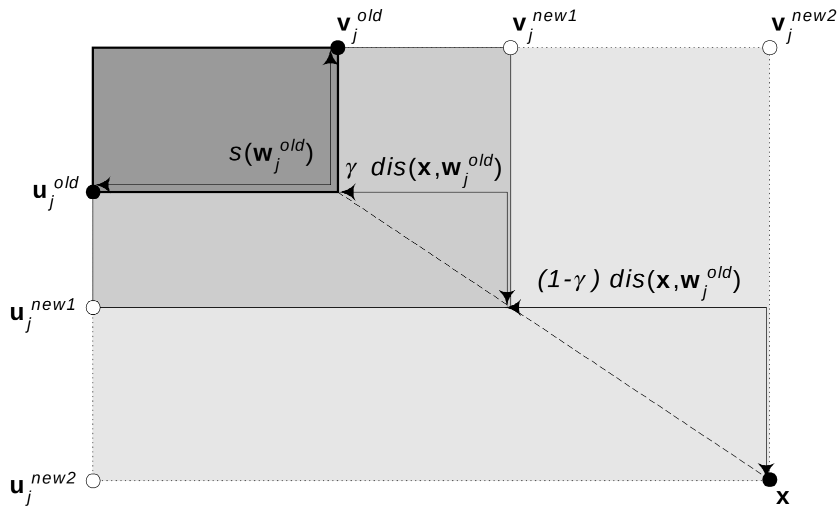

Figure 3: FA category updates - slow vs. fast learning.

After a category has been updated or created, training proceeds with the presentation of the next input pattern and so forth. A complete presentation of the entire training set is called a *list presentation* (*epoch*). When using fast learning we say that FAM has completed its learning task (converged), when after a complete list presentation no categories were created or updated.

Before we bring this section to a closure, it is important to underline the role of VT and CT in the operation of FAM. The VT is a category-filtering mechanism and is one of the components that implement the match-based aspect of FAM. If a category *j* does not satisfy the VT with respect to a specific input pattern **x**, this can be interpreted as "**x** does not match the geometrical characteristics of *j* and therefore should not select *j*". In the past it has been shown[5,6] that the CT, i.e., the competition against uncommitted nodes, is a category-filtering mechanism similar (but not identical) to the VT. In essence, both the CT and the VT act as novelty detection mechanisms and are able to detect patterns that are atypical with respect to whatever input has been experienced in the past by the FAM network.

3. FUZZY ART/ARTMAP WITH ADAPTIVELY WEIGHTED DISTANCES

In this paper we propose a modification to FAM, namely the adoption of a different distance measure, yielding a variant of FAM, which we call *Fuzzy ARTMAP with adaptively weighted distances* (FAMawd). For each category *j* that is competing for a pattern **x** let us define the vector quantities

$$\mathbf{l}_j = \mathbf{v}_j - \mathbf{u}_j . \tag{16}$$

$$\mathbf{d}_j = \mathbf{w}_j - \mathbf{x}^e \wedge \mathbf{w}_j . \tag{17}$$

The m^{th} component l_{jm} of $\mathbf{l}_j$ equals the length of j's representation region (hyper-rectangle) along the m^{th} axis. On the other hand, the m^{th} component d_{jm} of $\mathbf{d}_j$ equals the minimum L_1 distance of pattern $\mathbf{x}$ from j's representation region along the m^{th} axis. Obviously, it holds

$$s(\mathbf{w}_j) = \sum_{m=1}^{M} l_{jm} \ . \tag{18}$$

$$dis(\mathbf{x}, \mathbf{w}_j) = \sum_{m=1}^{M} d_{jm} \ . \tag{19}$$

Next, in Equation 20, we introduce the new distance measure that will be used in FAMawd for calculating the distance of a pattern $\mathbf{x}$ from the representation region of any category j.

$$dis(\mathbf{x}, \mathbf{w}_j \mid \lambda, ref) = \sum_{m=1}^{M} \frac{(1-\lambda) l_j^{ref} + \lambda}{(1-\lambda) l_{jm} + \lambda} d_{jm} \ . \tag{20}$$

where l_j^{ref} is a function of category j's lengths l_m, m=1…M and is called *reference length* for j. The new distance measure is parameterized by $\lambda \in (0,1]$. Notice that when λ=1 for any choice of l_j^{ref} the distance defined in Equation 20 becomes the regular city-block distance used in FAM, that is,

$$dis(\mathbf{x}, \mathbf{w}_j \mid 1, ref) = dis(\mathbf{x}, \mathbf{w}_j) \ . \tag{21}$$

Furthermore, it can be easily shown that for any choice of reference length l_j^{ref} the weighted L_1 distance dis($\mathbf{x}$,$\mathbf{w}_j$|λ,ref) is bounded as

$$\frac{M\lambda}{(M-1)\lambda + 1} dis(\mathbf{x}, \mathbf{w}_j) \le dis(\mathbf{x}, \mathbf{w}_j \mid \lambda, ref) \le \frac{1}{\lambda} dis(\mathbf{x}, \mathbf{w}_j) \ . \tag{22}$$

although the above bounds are not necessarily tight. Typical choices for l_j^{ref} can be l_j^{max} and l_j^{avg}, which are defined below:

$$l_j^{\max} = \max_{m=1..M} \{ l_{jm} \}. \tag{23}$$

$$l_j^{avg} = \frac{1}{M} \sum_{m=1}^{M} l_{jm} \ . \tag{24}$$

Also, we designed FAMawd to use the following CMF:

$$\rho(\mathbf{w}_j \mid \mathbf{x}) = \frac{D - s(\mathbf{w}_j) - dis(\mathbf{x}, \mathbf{w}_j \mid \lambda, ref)}{D} \ . \tag{25}$$

Moreover, the expressions for Weber Law and Choice-by-Difference CCF in FAMawd take the form displayed below:

$$T(\mathbf{w}_j \mid \mathbf{x}) = \frac{D - s(\mathbf{w}_j) - dis(\mathbf{x}, \mathbf{w}_j \mid \lambda, ref)}{D - s(\mathbf{w}_j) + a} \ . \tag{26}$$

$$T(\mathbf{w}_j \mid \mathbf{x}) = D(2-a) - as(\mathbf{w}_j) - dis(\mathbf{x}, \mathbf{w}_j \mid \lambda, ref) \ . \tag{27}$$

Equations 25, 26 and 27 use a network parameter D>0, such that $\rho(\mathbf{w}_j|\mathbf{x})\geq 0$ for every category j and pattern **x**. In our design we have used the following values of D, when l_j^{max} (Equation 28) and l_j^{avg} (Equation 29) are used for the calculations of weighted distance:

$$D = 1 + \frac{M-1}{\lambda}. \tag{28}$$

$$D = \begin{cases} M\left[1+\frac{1-\lambda}{4\lambda}\right] & \text{if } M \text{ even} \\ M\left[1+\frac{1-\lambda}{4\lambda}\left(1-\frac{1}{M^2}\right)\right] & \text{if } M \text{ odd} \end{cases}. \tag{29}$$

Notice that the CMF and CCF expressions in Equations 25, 26 and 27 simplify to the ones of FAM's depicted in Equations 10, 5 and 6, when λ=1. Aside from the different CMF and CCF expressions, the operations and features of FAMawd are identical to the ones of FAM. Therefore, we can state that FAM is a special case of the FAMawd network.

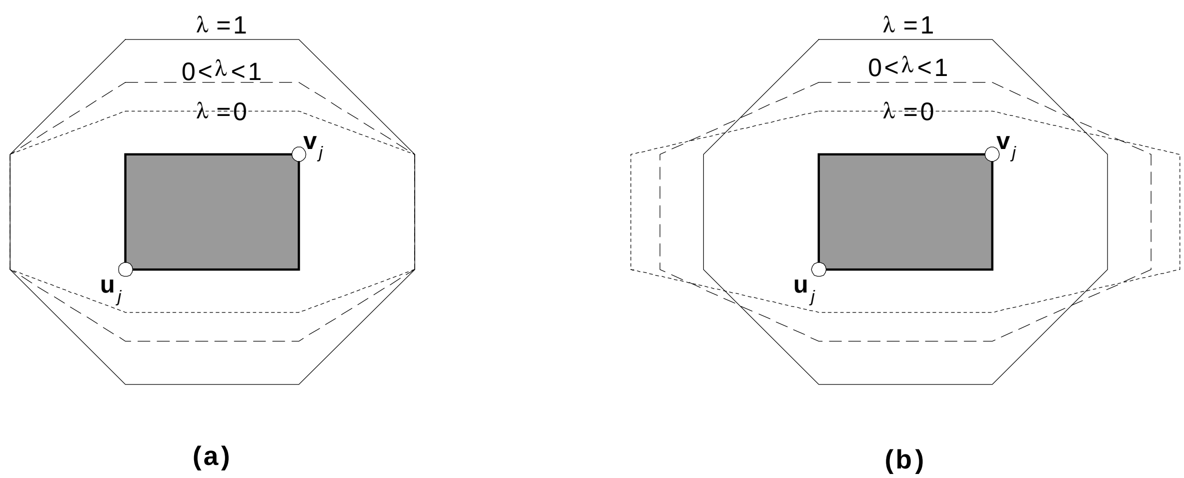

Figure 4: Contours of constant weighted L_1 distance for different values of λ.
In subfigure (a) $l_j^{ref} = l_j^{max}$ is used and in subfigure (b) $l_j^{ref} = l_j^{avg}$.

The effect of using the weighted L_1 distance of Equation 20 for a 2-dimensional category in FAMawd is presented in Figure 4. The contours that are plotted for 3 different values of λ around the category's representation region correspond to the points of the input domain that are located at a constant weighted L_1 distance from the category. It should become apparent that categories in FAMawd will tend to expand during training along the axis, for which they feature maximum length. In other words, categories in FAMawd will tend to preserve their already formed direction throughout the training phase. It is also important to point out that the newly introduced distance, as defined in Equation 20, features weights that adapt according to the particular shape of a category after it becomes updated pronouncing this way the preservation of direction.

4. EXPERIMENTAL RESULTS

In order to compare the FAMawd and the FAM architectures we have performed preliminary experiments on two different data sets. The first data set consists of 2200 patterns obtained from the abalone[7] database. The first 200 patterns were used for training, the next 1000 patterns for validation, and the last 1000 for testing. The second experiment is the "circle in a square" problem, where the circle is located at the center of the square and occupies half the square's area. The data set consists of 10200 patterns. The first 200 patterns were used for training, the next 5000 patterns for validation, and the last 5000 for testing. For the experiments we considered 30 orders of pattern presentation and 20 different values (0,0.05,…, 0.95) for the vigilance parameter ρ. Also, for the parameter α we considered 9 different values (0.1, 0.2, …, 0.9) for the CBD choice function and 7 values (0.0001 0.001 0.01 0.1 1.0 10.0 100.0) for the Weber Law choice function. Finally, we used 11 different values (0.5, 0.55, ..,1) for the parameter λ. Let us note that in all of our experimentations we assumed that $w_u \rightarrow \infty$.

We have used cross validation for each order of pattern presentation. More specifically, one FAM network and one FAMawd network are selected for each order. These are the ones that maximize the percent correct classification (PCC) on the validation set with respect to the different architecture parameters. The networks chosen via the cross validation procedure could be the ones that maximize the PCC on the validation set with respect to both the different architecture parameters and the different orders of pattern presentation. However, we have treated the two (architecture parameters and orders of pattern presentation) separately. The justification is that the PCC on the validation set can be sufficiently examined for the different parameters if a large set of parameter values is chosen. In contrast, there can be an exceptionally large number of different orders of pattern presentation, so that it is unrealistic to use cross validation for all possible orders.

Therefore, in order to present a fair comparison between FAM and FAMawd we have plotted the PCC on the test set versus the corresponded PCC on the validation set and also the number of nodes created versus the corresponded PCC on the validation set. These plots for both experiments are shown in Figures 5 and 6. Figure 5 shows the results for the experiments on the abalone data set. Figures 5(a) and 5(b) present respectively the PCC on the test set and the number of nodes created versus the PCC on the validation set for all 30 orders of pattern presentation using CBD. Figures 5(c) and 5(d) present the corresponding results, when using WL. In each plot, we also show the two lines that best fit the points that correspond to FAM and FAMawd. From Figures 5(b) and 5(d) we can conclude that the number of nodes created for orders that have given a high PCC on the validation set is generally smaller for FAMawd than for FAM. The latter is verified by the more negative slope of the line that best fits the points for FAMawd in both plots. For instance, the average number of nodes for the orders of pattern presentation that have given the best 7 PCC scores on the validation set is 128 for FAM and for 108 FAMawd, when using CBD, and 78 for FAM and 55 for FAMawd, when using WL. On the other hand, there is no specific pattern for the PCC on the test set, and the two architectures seem to have similar performance. For instance, the average PCC for the orders of pattern presentation that have given the best 7 PCC scores on the validation set is 46.1% for FAM and for 46.2% FAMawd, when using CBD, and 46.3% for FAM and 46.1% for FAMawd, when using WL.

Similar conclusions can be drawn from the results presented in Figure 6 for the "circle in a square" experiment. Again, from Figures 6(b) and 6(d) we can conclude that the number of nodes created for those orders that have given a high PCC on the validation set is smaller for FAMawd than for FAM. For instance, the average number of nodes for the orders of pattern presentation that have given the best 7 PCC scores on the validation set is 79 for FAM and for 69 FAMawd , when using CBD, and 61 for FAM and 42 for FAMawd, when using WL. Again, there is no specific pattern for the PCC on the test set, and the two architectures seem to have similar performance, with a slightly better PCC for the FAMawd. The average PCC for the orders of pattern presentation that have given the best 7 PCC scores on the validation set is 96.3% for FAM and for 96.5% FAMawd, when using CBD, and 96.1% for FAM and 96.6% for FAMawd , when using WL.

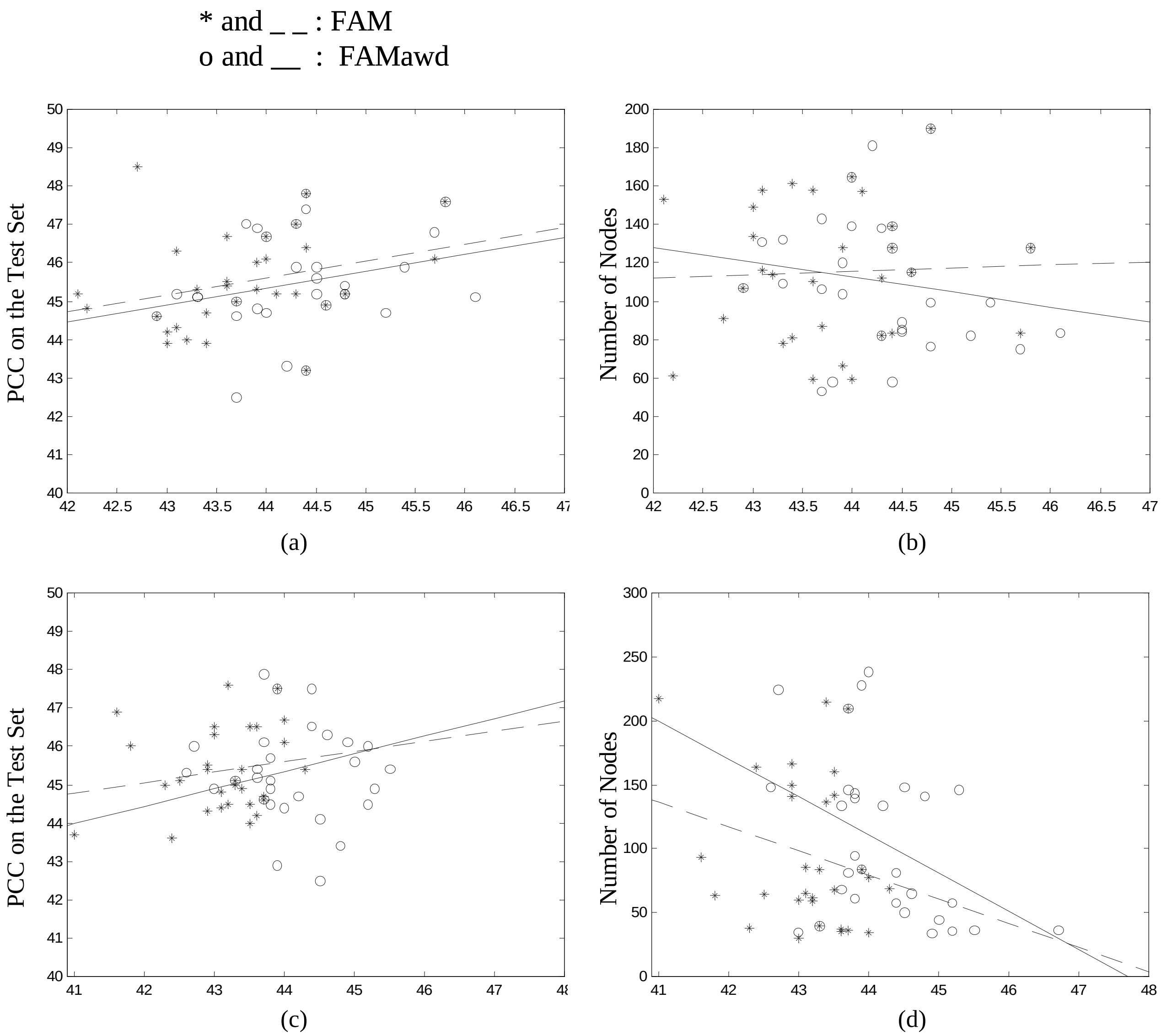

Figure 5. Results for the abalon data set: (a) PCC on the test set versus PCC on the validation set (choice by difference), (b) Number of nodes created versus PCC on the validation set (choice by difference), (a) PCC on the test set versus PCC on the validation set (Weber's Law), (a) Number of nodes created versus PCC on the validation set (Weber's Law).

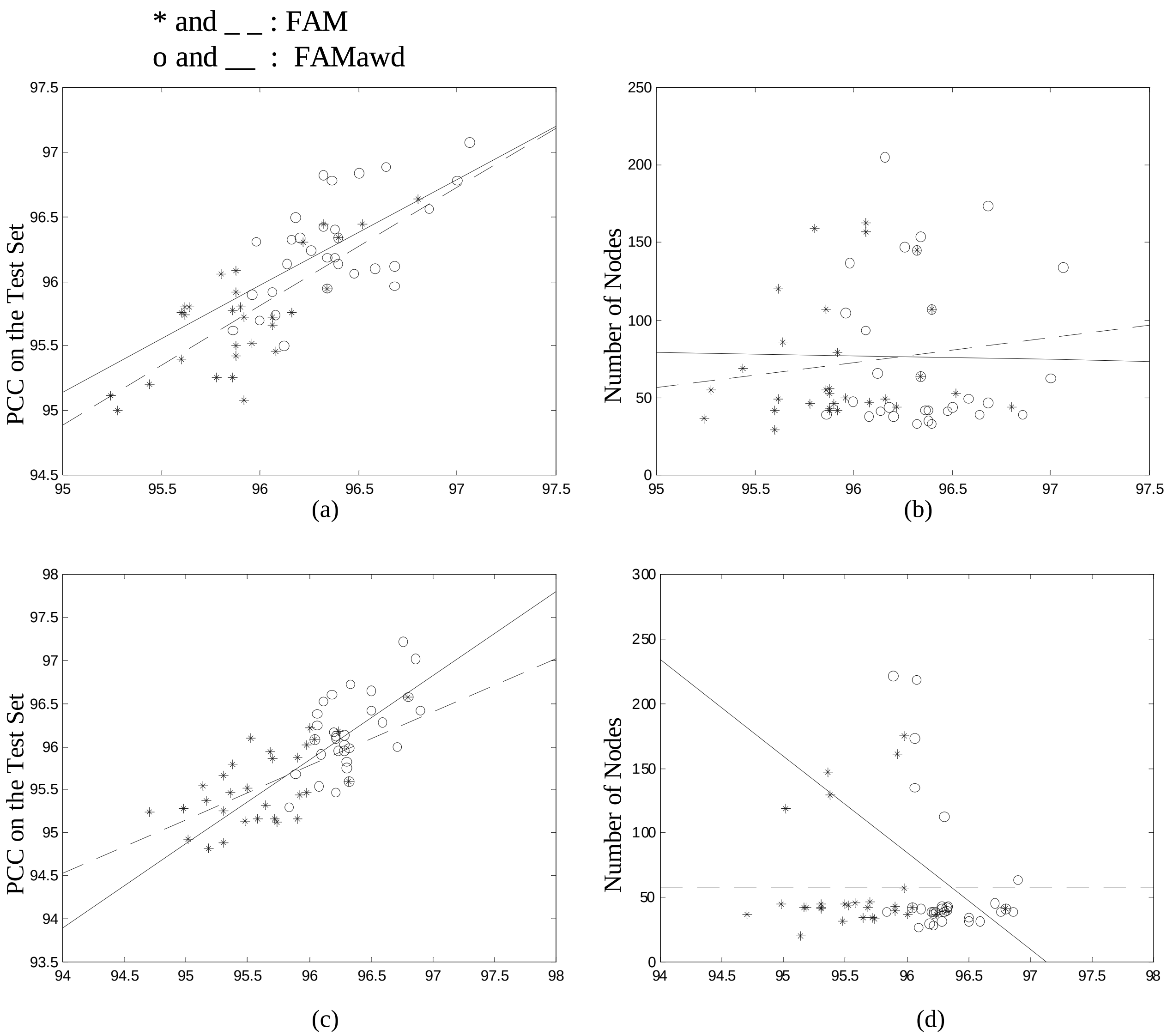

Figure 6. Results for the “circle in a square” problem: (a) PCC on the test set versus PCC on the validation set (choice by difference), (b) Number of nodes created versus PCC on the validation set (choice by difference), (a) PCC on the test set versus PCC on the validation set (Weber’s Law), (a) Number of nodes created versus PCC on the validation set (Weber’s Law).

5 SUMMARY AND CONCLUSIONS

In this paper, we proposed a modification of the Fuzzy ARTMAP (FAM) neural network, namely, the Fuzzy ARTMAP with adaptively weighted distances (FAMawd) neural network. In FAMawd we substitute the regular L_1-norm with a weighted L_1-norm to measure the distances between categories and input patterns. The distance-related weights are a function of a category’s shape and allow for bias in the direction of a category’s expansion during learning.

Moreover, the modification to the distance measurement is proposed in order to study the capability of FAMawd in achieving more compact knowledge representation than FAM, while simultaneously maintaining good classification performance. For a special parameter setting FAMawd simplifies to the original FAM, thus, making FAMawd a generalization of the FAM architecture. We also presented an experimental comparison between FAMawd and FAM on two benchmark classification problems in terms of generalization performance and utilization of categories. The results showed that the proposed FAMawd neural network creates, in general, smaller architectures than the original FAM neural network for those orders of pattern presentation that result in a large percent of correct classification (PCC) on the validation set. This is a significant result, since networks with high PCC on the validation set are more likely to be chosen as the final candidates. In summary, our results illustrate FAMawd's potential to exhibit low memory utilization, while maintaining classification performance comparable to FAM.

REFERENCES

1. G.A. Carpenter, S. Grossberg and D.B. Rosen, "Fuzzy ART: Fast stable learning and categorization of analog patterns by an adaptive resonance system", *Neural Networks*, 4(6), pp. 759-771, 1991.
2. S. Grossberg, "Adaptive pattern recognition and universal encoding II: Feedback, expectation, olfaction, and illusions", *Biological Cybernetics*, **23**, pp. 187-202, 1976.
3. G.A. Carpenter, S. Grossberg, N. Markuzon, J.H. Reynolds and D.B. Rosen, "Fuzzy ARTMAP: A Neural Network Architecture for Incremental Supervised Learning of Analog Multidimensional Maps", *IEEE Transaction on Neural Networks*, **3**(5), pp. 698-713, 1992.
4. G.A. Carpenter and M.N. Gjaja, "Fuzzy ART choice functions", *Proceedings of the World Congress on Neural Networks (WCNN'94)*, **5**, pp. 133-142,1994.
5. G.C. Anagnostopoulos and M. Georgiopoulos, "New Geometrical Concepts in Fuzzy-ART and Fuzzy-ARTMAP: Category Regions", *Proceedings of the IEEE-INNS-ENNS International Joint Conference on Neural Networks (IJCNN '01)*, Washington, Washington D.C., **1**, pp. 32-37, 2001.
6. G.C. Anagnostopoulos, *Novel Approaches in Adaptive Resonance Theory for Machine Learning*, Doctoral Dissertation, University of Central Florida, Orlando, Florida, 2001.
7. C.L. Blake and C.J. Merz, UCI Repository of machine learning databases [http://www.ics.uci.edu/~mlearn/MLRepository.html], Department of Information and Computer Science, University of California, Irvine, California, 1998.