# A Review of MF-ARTMAP Toward to Improvement Classification Accuracy using Simulated Annealing

Jaroslav Ondo[1], Patrik Sabol[1], Peter Sinčák[1], Pitoyo Hartono[2]
[1]Technical University of Kosice (TUKE)/Department of Artificial Intelligence and Cybernetics, Košice, Slovakia
School of Engineering, Chukyo University, Nagoya, Japan
{jaroslav.ondo, patrik.sabol, peter.sincak}@tuke.sk, hartono@ieee.org

***Abstract* – This paper deals with an MF ARTMAP neural network. We study its behavior while training with different data sets and using different parameters. It gives us better knowledge of its strong and weak points. Subsequently, we focus on alleviation of weak points and improvement of strong points like the utilization of a one-shot learning, an incremental ability of the network without forgetting the already obtained knowledge or post-processing of information stored in the form of the transparent internal structure of identified clusters and classification classes. We have shown the incrementality of this neural network. As for the weak part of the MF ARTMAP algorithm, we try to increase the generalization ability by adopting Simulated Annealing method to find the best shape of membership functions with the best possible ratio between generalization of the neural network and its classification performance. Using simulated annealing algorithm, we optimize network's parameters namely the membership function's shapes of fuzzy clusters in the feature space. Subsequently, we compare classification accuracy of MF ARTMAP with and without parameters optimization, as well. Moreover, we compare against the classification precision of the Multi-Layer Perceptron (MLP) using benchmark data sets, with the aim to get a relevant image of the overall MF ARTMAP efficiency beside the well-known and frequently-used algorithm, like the MLP.**

***Keywords – classification, neural network, MF ARTMAP, optimization, semantics, Simulated annealing***

## I. Introduction

Since 1943, when Warren McCulloch and Walter Pitts presented their paper on how neurons might work [1], the era of artificial neural networks has started. This seminal paper has unleashed an avalanche of development in the field of artificial neural networks (ANNs). Since then, many kinds of neural networks with various topologies have been developed, each suitable for solving a particular task [2]. Nevertheless, almost all types of neural networks suffer from catastrophic forgetting [3] of previously learned findings when requested to acquire new knowledge or when the learning system should follow changing environment.

Stephan Grossberg in [4] first formally described the problem and called it a stability-plasticity dilemma. In general, the stability-plasticity dilemma formulates requirements for learning systems. Each learning system should be stable in response to known inputs but also, should be plastic enough to recognize and learn new inputs.

A possible solution to the plasticity-stability dilemma is the Adaptive Resonance Theory (ART) [5], initially introduced by Grossberg and Carpenter. This theory embeds competitive and learning model into the self-organized structure [6]. Such self-organized structure can be a neural network, which can recognize and classify an arbitrary sequence of input samples in real time. The ART self-organizes its topology and generates stable recognition codes during the learning process.

Although the implementation of the ART theory was intended as a single neural network, with time, many modifications and improvements of the basic ART network, like ART 2 [7], ART 2-A [8], Fuzzy ART [9] were developed. Their common characteristic is that all of them are recurrent self-organized neural networks, trained in an unsupervised manner. On the other hand, ARTMAP is a subgroup of ART networks, which can be trained in a supervised manner.

Studies in [6][10][11] and [12], characterize Membership Function (MF) ARTMAP, as a supervised recurrent neural network, which has qualities like incremental learning or transparent network structure that are rare for neural networks. Thanks to them, the MF ARTMAP seems to be a promising solution for the problem of catastrophic forgetting. Those papers highlight a possibility of one-shot learning and denote that MF ARTMAP is not a black box in the strict sense. It is related to the core functionality and knowledge representation of the MF ARTMAP, which clusters input samples in the feature space. Moreover, the internal structure of the network is constantly updated and reflects the structure of the data's clusters in the feature space. Hence, we can extract interesting information about clusters and classification classes from the network structure in sematic expressions. The MF ARTMAP seems to be an interesting network, which can provide several valuable features. Unfortunately, there is no sufficient theoretical study of MF ARTMAP and its mathematical background is not sufficiently studied. Therefore, we decided to study the MF ARTMAP algorithm.

In the second section, we describe the MF ARTMAP's topology and the learning algorithm. In the next section, we provide an example of the incremental ability of the MF ARTMAP network in comparison with MLP. The fourth section is focused on the explanation of performed trials, identified issues, and their solutions. The fifth section unveils further potentials of the neural network towards to semantics extraction. Finally, we will conclude this work.

## II. MF ARTMAP TOPOLOGY AND LEARNING ALGORITHM

As the name implies, MF ARTMAP belongs to the group of ARTMAP neural networks and poses a symbiosis of fuzzy sets theory and ART theory. The core functionality of this network is a clustering of input patterns in the feature space. Fuzzy clusters (sets)[1] consist of clustered inputs in the feature space as illustrated in Fig.1. Fuzzy class consists of one or several fuzzy clusters which means that we can compute a value of membership $\mu_A(x)$ of each input sample $x$ to each fuzzy cluster $A$. The maximum of those membership values represents the membership of the input to the associated class. Although it is possible to use arbitrary membership function, the Cauchy-like bell-shaped membership function is frequently used.

We can define fuzzy cluster using a fuzzy relation with three parameters for each dimension of each cluster ($k$). Namely center ($X_{s,k}$), variance ($E_k$) and cluster's kernel width ($F_k$). If the input vector is multidimensional, then mentioned parameters are defined as vectors, which elements are defined for an appropriate dimension of the cluster. MF ARTMAP learning algorithm adjusts parameters of the fuzzy relation which results in the modification of the membership function's shape. The shapes of the membership functions are adapted so that they cover all objects in the feature space belonging to the class associated with each of them.

$$\mu_k(\overline{x_i}) = \frac{1}{1 + \left| \dfrac{k}{\sum_{j=1}^{m} f_{k,j}(x_{i,j})} - 1 \right|} \tag{1}$$

Equation (1) shows the membership value of the $i$-th, $m$-dimensional input vector $\overline{x_i}$ to the $k$-th fuzzy cluster. Index $j$ is a dimension index and dimensionality of fuzzy cluster is the same as dimensionality of the input vector. The function, $f_{k,j}$ represents the $j$-th dimensional membership of the k-th fuzzy cluster define as follows.

$$\boldsymbol{f_{k,j}(x_{i,j})} = \frac{\mathbf{1}}{\mathbf{1} + \left( \dfrac{\left| x_{s_{k,j}} - x_{i,j} \right|}{E_{k,j}} \right)^{F_{k,j}}} \tag{2}$$

Wherein $x_{i,j}$ is a $j$-th dimension of the input vector, $x_{s_{k},j}$ is center of $k$-th fuzzy cluster and its $j$-th dimension. $E_{k,j}$ and $F_{k,j}$ are parameters of the $k$-th fuzzy cluster in the j-th dimension. [6] $E_{k,j}$ is a variance of the $j$-th dimension of the $k$-th fuzzy cluster. The increase of the $E_{k,j}$ parameter causes the widening of the membership function in that dimension. Increasing of the $F_{k,j}$ parameter is equivalent to increasing the steepness of the membership function. Fig. 1. presents the structure of the MF ARTMAP neural network. It consists of four neuron layers as follows:

1. **Input layer** propagates an input pattern to the comparison layer, where it is possible to perform normalization or standardization of the inputs. The number of neurons in the input layer is the same as the dimensionality of the input vector.

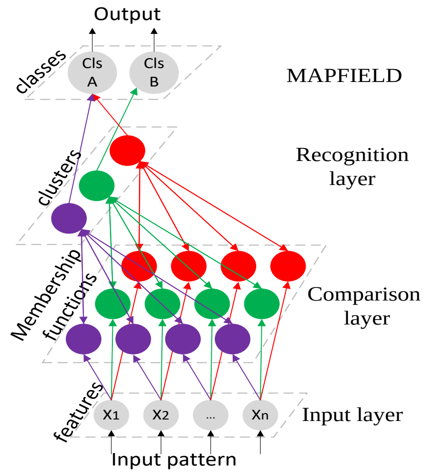

Fig. 1 A topology of the MF ARTMAP network

2. **Comparison layer** computes a partial membership function for each dimension of each known fuzzy cluster according to equation (2). The neurons in this layer are aligned in two-dimensional grid, in which columns represent the fuzzy clusters while rows represent cluster's dimensions.

3. **Recognition layer** summarizes partial membership values to the total value of membership function of each fuzzy cluster according to (1). Each neuron in this layer represents one fuzzy cluster. Once the learning algorithm computes the total membership value, this value is compared with a threshold value set by the user. If the total membership value exceeds the threshold, then the input vector is considered to be similar enough to the currently investigated fuzzy cluster, and this cluster is suitable for next processing in the Mapfield layer. Otherwise, the investigated fuzzy cluster does not describe the input correctly, and the signal from this neuron is not propagated to the next layer.

   If no cluster is assigned to the input, then the learning algorithm modifies the network's structure by adding a new neuron to the recognition layer and new row of neurons to the comparison layer, which is equivalent to a definition of a new fuzzy cluster which is centered in the input sample.

4. **Mapfield** contains the same number of neurons as the number of classes. Since each class consists of unification of fuzzy clusters, the membership value of the fuzzy class is equal to the maximum membership value of its fuzzy clusters. The output from the MF ARTMAP network can be in the form of a vector, which consists of membership values of the input pattern to the respective fuzzy class. The second possibility and usually more common output is the name of the fuzzy class with the maximum value of membership function. If the output class does not correspond to the desired output of the input pattern, the algorithm adds a new fuzzy cluster according to the approach mentioned in point 3. Otherwise, the adaptation

[1] In this paper, we consider a fuzzy class or cluster and fuzzy set as equivalent terms.

of the parameters ($x_{s_k}$, $E_k$ and $F_k$) occurs. Since the recurrent synaptic connection between comparison and recognition layer encode those parameters, the adaptation procedure adjusts those links.

$$X_S^N = X_S^O - \frac{1}{q}(X_S^O - X) \tag{3}$$

$$E^N = E^O - \frac{1}{q}(E^O - (X_S^O - X)^2) \tag{4}$$

$$F^N = F^O - \frac{1}{q}(F^O - (X_S^O - X)^2) \tag{5}$$

Wherein $q$ is the number of all input patterns covered by the fuzzy cluster; $O$ denotes the old value of the variable, $N$ refers to the adjusted parameters, and $X$ is an input sample.

## III. Incremental Learning of the MF ARTMAP

From the nature of the ART-like networks follows that incremental learning ability without catastrophic forgetting. Although the incremental ability should be obvious due to the self-organization nature of the network structure, it is not well covered in the existing literatures. Hence, we dedicate this section to provide such experiments. We proposed simple trials, in which we trained MF ARTMAP and Multi-Layer Perceptron (MLP).We trained and evaluated both neural networks in two steps. For those purposes, we used IRIS data [13], which are often used as a benchmark for classification. IRIS data contains 150 input vectors that belong to three classes. We created two data sets. The first data set contained 100 input patterns, which belong to two classes. We split this data set for training set (70 input patterns; 35 for class A and 35 for class B) and testing set (30 input patterns; 15 for class A and 15 for class B). Then, we used the rest (50 vectors), which belong to the class C, for the second data set. We split this data set into training and testing sets, as well. The training set consisted of 35 inputs for class C. Test set contained 15 patterns of class C and also all patterns from the first test set. So, the total number of instances in the second test set is 45 (15 instances of class A, 15 for class B and 15 for class C).Here, we attempt to show that the MF ARTMAP can correctly classify all input patterns without forgetting the already gained knowledge. The experiment was executed as follows:

1. Training MF ARTMAP and MLP with the 1st training set.
2. Evaluating both networks using the 1st testing set and visualizing the results using contingency tables.
3. Usage of the 2nd training set for the incremental training of the already trained MF ARTMAP and MLP neural networks.
4. Re-evaluating both neural networks with the 2nd testing set and visualizing the results using contingency tables.

We expected high classification accuracy for both networks after training by patterns from the 1st training set. However, in the case of MLP, the 2nd training causes the MLP network to be able to correctly classify only the class $C$.

The generated confusion matrices (Table 1 and 2) agree with our hypothesis, and thus, we have demonstrated the incremental ability of the MF ARTMAP learning without forgetting the already gained knowledge. MF ARTMAP can be trained in incremental nature due to its self-organization capacity and establishment of new clusters. As can be seen from Tab. 1, the MF ARTMAP before incremental learning, had 23 clusters. During the incremental training, it gained new clusters and finally had 37 clusters.

Table. 1 Contingency tables for MF ARTMAP after training by the 1st training set (left) and after incremental learning (right).

| Real\ Computed | A | B | Unknown | Σ |
|---|---|---|---|---|
| A | 15 | 0 | 0 | 15 |
| B | 0 | 15 | 0 | 15 |
| Σ | 15 | 15 | 0 | 30 |

23 clusters

| Real\ Computed | A | B | C | Unknown | Σ |
|---|---|---|---|---|---|
| A | **15** | 0 | 0 | 0 | 15 |
| B | 0 | **14** | 1 | 0 | 15 |
| C | 0 | 0 | **15** | 0 | 15 |
| Σ | 15 | 14 | 16 | 0 | 45 |

37 clusters

Table. 2 Contingency tables for MLP network after training by the 1st training set (left) and after incremental learning (right).

| Real\ Computed | A | B | Σ |
|---|---|---|---|
| A | **15** | 0 | 15 |
| B | 0 | **15** | 15 |
| Σ | 15 | 15 | 30 |

| Real\ Computed | A | B | C | Σ |
|---|---|---|---|---|
| A | **0** | 15 | 0 | 15 |
| B | 0 | **0** | 15 | 15 |
| C | 0 | 0 | **15** | 15 |
| Σ | 0 | 15 | 30 | 45 |

## IV. Description of Identified Issues and Solutions

In this section, we discuss MF ARTMAP from the empirical point of view. We observe on how the MF ARTMAP behaves in different situations, against various inputs and various settings.

We have prepared several data sets, which we have used for a measurement of classification performance of the MF ARTMAP algorithm. Those data sets contain 2 or 3 dimensional artificially generated data. Those data follow the Normal distribution for each cluster. Each dataset consists of 2 – 4 clusters, belonging to 2 or 3 classes. We have simulated the different overlap between clusters in each data set, and we have used a different number of samples for each cluster and each class (200 – 2000 patterns/class). Since those datasets are easy to visualize, they are suitable for full understanding of MF ARTMAP procedure and can help us to unveil situation in feature space easily. MF ARTMAP classification process accuracy assessment principles were used according [14].

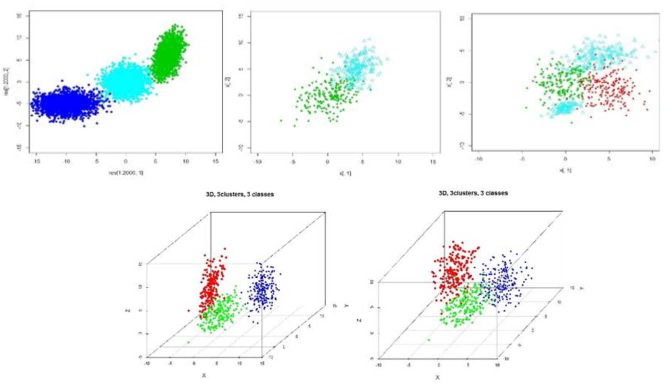

Fig. 2 Feature space examples of generated 2 and 3-dimensional data sets.

Since equations (4) and (5) are the same, we focused on verification of this fact. It is necessary to prove the correctness of default formulas, which are responsible for adjusting parameters of found clusters. We need to emphasize that parameters of fuzzy clusters are adjusted iteratively.

During the verification of the mentioned formulas, we discovered that equation (4), which is responsible for an iterative computation (adjustment) of $E_k$ parameter gives slightly different results, as we expected in comparison with the case when we did not compute the standard deviation iteratively. Nevertheless, the equation (4) is mathematically correct, we empirically found equation modification, which works exactly as we expected. Adjusted equations are as following:

$$X_S^N = \frac{(q \cdot X_S^O + X)}{q+1} \tag{6}$$

$$(X_S^N)^2 = \frac{(q \cdot (X_S^O)^2 + X^2)}{q+1} \tag{7}$$

$$(E^N)^2 = (X_S^N)^2 - (X_S^N \cdot X_S^N) \tag{8}$$

Wherein $q$ is the number of all input patterns covered by a fuzzy cluster; $O$ means the old value of a variable, $N$ is a new parameter value.

Equations after modifications give the expected results in comparison with non-iterative computation. However, there is still the problem of computing $F$ parameter.

The experiments showed that situations in which the MF ARTMAP is unable to classify all patterns from the training test set, arise. This is due to the non-optimal value of $F$ parameter.

Figure 3 presents an influence of $F$ parameter on the shape of the Cauchy-like membership function; wherein $x$-axis is an input value for one dimension and $y$-axis means the membership value. In the (a) part, the parameter $F$ is set to values 1 or 2 for different clusters. It caused the value of membership belong to the interval from 0 to 1, and each cluster is relatively narrow. There exist sub-spaces in the feature space, where all clusters have their membership values equal to zero. Training patterns do not cover those places, and so the algorithm could not create a cluster there. If the test pattern belongs to this subspace, then the algorithm will be unable to classify it; the output is unknown class.

For example, in Fig. 3 (a), the interval from -3 to -1 is the typical of an uncovered area. In selected case, it is better to have a large training set, with patterns which can cover the entire feature space by many narrow, distinct clusters. Then, we can reach a high classification precision and alleviate a problem of overlapping clusters, but in the test set are many patterns that the algorithm does not know to classify.

We set the $F$ values in Fig. 3 (b) to 0.2 or 0.1. Such setting causes that membership values not to cover the entire interval [0;1]. It causes that clusters are wide with very slight drop of the membership value. Using mentioned settings, we can cover the entire feature space, but usually at the cost of the classification precision. For instance an input pattern, whose value is -3 should be classified to the red cluster because its center is the closest to the pattern of entry. However, this pattern will be classified into the green cluster because there is the greatest membership value. In this case, we do not need many training patterns to cover the entire feature space. Not every training example is classified correctly, as well. The algorithm can classify all testing patterns, but mostly into the wrong class. The clusters overlapping problem occurs, as well.

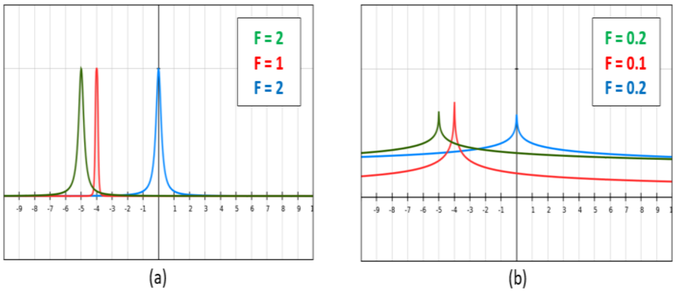


Fig. 3 $F$ parameter influence on the shape of the Cauchy-like membership function.

We need to find a balance between those scenarios with the goal to combine high classification precision of the first case with a generalization ability of the second case. Therefore, we decided to employ an optimization algorithm in order to optimize fuzzy cluster parameters. Our goal is to find an optimal value of the $F$ parameter of each fuzzy cluster and each dimension. It is possible to use any optimization algorithm, for instance, evolutionary or genetic algorithms, hill climbing methods or gradient methods, we decided to employ a well-known simulated annealing [15] because of its simplicity and implementation clarity.

Simulated annealing is suitable for utilization in wide range of optimization problems; the only requirement is a definition of a cost function. We defined cost function as the classification accuracy, which we computed as a Kappa coefficient [16][17] from the contingency table. It means, that each time, when the clusters parameters were changed we evaluate the classification performance using the test set, create confusion matrix and compute Kappa coefficient.

In the table 3, we summarized the average classification precision of MF ARTMAP without simulated annealing optimization, with optimization and comparison with MLP against some benchmark test data [13].

| Data set | MF Artmap | MF Artmap with optimization | MLP |
|---|---|---|---|
| IRIS | 90% | ***100%*** | 97% |
| THYROID | 85% | 86% | ***98%*** |
| FERTILITY | 75% | 91% | ***95%*** |
| WINE | 83% | 90% | ***97%*** |
| PIMA | 60% | ***81%*** | 63% |
| BUPA | 61% | ***72%*** | 68% |
| ECOLY | 65% | 80% | ***84%*** |
| BREAST CANCER | 92% | 97% | ***98%*** |

Table 3. Comparison of classification precision between MF ARTMAP without and with optimization and with MLP. The best result for each data set is bold italic.

All values in the table 3. are averaged since the classification precision of the MF ARTMAP is dependent on the order of training samples. Thus, we trained MF ARTMAP at least ten times and computed the average precision. As the results show, the optimization of clusters parameters using the simulated annealing in the MF ARTMAP network increases classification accuracy. Another effect of the optimization is that the number of failed classifications (into the Unknown class) dramatically decreased. As an example we provide contingency tables for THYROID dataset (A,B,C classes), when the MF ARTMAP was trained without optimization and with simulated annealing optimization.

Table. 4 Contingency tables for MF ARTMAP trained by THYROID data without optimization

| Real\ Computed | A | B | C | Unknown | Σ |
|---|---|---|---|---|---|
| A | 62 | 2 | 6 | 81 | 151 |
| B | 0 | 13 | 0 | 21 | 34 |
| C | 0 | 0 | 7 | 23 | 30 |
| Σ | 62 | 15 | 13 | 125 | 215 |

Table. 5 Contingency tables for MF ARTMAP trained by THYROID data with simulated annealing optimization

| Real\ Computed | A | B | C | Unknown | Σ |
|---|---|---|---|---|---|
| A | 143 | 6 | 2 | 0 | 151 |
| B | 0 | 24 | 0 | 10 | 34 |
| C | 2 | 0 | 14 | 14 | 30 |
| Σ | 145 | 30 | 16 | 24 | 215 |

Moreover, the comparison with the MLP showed that the MF ARTMAP with optimization has similar performance with that of MLP. However, the MF ARTMAP has clear advantage if we consider that MF ARTMAP allows incremental learning without catastrophic forgetting.

Despite the issues described in the previous section, our experiments uncovered various benefits and further potential of this algorithm. The most valuable are a possibility of one-shot incremental learning and output in the form of transparent information concerning to relations between the observed point in the feature space and all known classes. One-shot learning means that the learning process is not iterative. In other words, MF ARTMAP neural network can gain appropriate knowledge in one step.

Furthermore, unlike other neural networks, the MF ARTMAP is not a black box in the strict sense. Since the algorithm can compute the membership value of the input to each class, the information about the clusters, their centers, and shapes of their membership functions must be stored in the network. This information can be utilized for post-processing as follows.

### A. Generalization and number of clusters in the feature space

If the number of clusters is large, the classification error can be low, but processing of the network can be very slow because the algorithm has to investigate a large number of clusters. Contrariwise, if the categorization error increases, the reason for this can be a lower number of clusters. [6] The aim of manipulation with the number of clusters is to find an optimal number of clusters. It is possible to do in two ways:

*1) Merging clusters*

Merging decreases the total number of clusters. Once, the learning is terminated, we can post-process information about the clusters. The post-processing searches clusters belonging to the same class, and if they are close they can be merged. This way, we can decrease the number of all clusters and speed up the MF ARTMAP. In this case we need to compute a center of a new cluster and new $E$ and $F$ parameters, which can hold all patterns of the original two clusters as selected in Fig. 4 (a). In the picture $x$ is a dimension of the input and $y$ is a membership value.

*2) Learning in several cycles*

Figure 4 shows that not whole feature space is covered by clusters. This phenomenon is possible to eliminate by learning in several cycles. Whereas the MF ARTMAP adopts the incremental algorithm, the algorithm has a capability to creating new clusters in the feature space. Therefore, we can train it several times, each time by the same train set, but with different order of the training patterns. This approach allows us to fill valleys between clusters, at the cost of the one-shot learning.

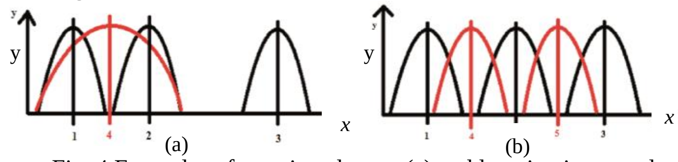


Fig. 4 Examples of merging clusters (a) and learning in several cycles (b) [6].

### B. Extraction information about clusters and classes

The structure of clusters in the feature space and also classes distribution for each cluster are known. It is a base for high-level processing, giving auxiliary output about the clusters, classes, their structures or similarities and dissimilarities between classes. The post-processing can offer information about one particular class (Intra-class knowledge) as well as the relation between several classes (Inter-class knowledge). Intra-class knowledge usually expresses a dispersion of clusters, which belong to one class in the feature space. Contrariwise, Inter-class knowledge extracts relation between clusters by using a computation of coverage between clusters. A suitable method for computing coverage is a Jeffries-Matsushita distance. This way, it is possible to obtain information about class independence and similarity. The difference between them is that class independence investigates independence of one class from all others classes while class similarity examines every time the similarity between two classes.

### C. Interpretation of multivalue outputs as results of classification process

The MF ARTMAP computes the membership values of the input to each fuzzy class and fuzzy class is union of fuzzy clusters. This information can be considered as a multivalued output and can be used for the improvement of classification. Classification accuracy is assessed with contingency table.

Since each fuzzy class in the MF ARTMAP consists of unified fuzzy clusters, we can compute the membership value to each fuzzy cluster. It allows us for generating a contingency table for each cluster. Subsequently, by using the contingency tables we can make a statement about a confidence of the output. Table 6. shows two contingency tables. Each of them was generated for different clusters, and each of those two clusters classifies the same class. The cluster associates with the class "0" and depicted by contingency table (a) have classified correctly for every input. Therefore, if we can state that the output from this cluster is a class "*0"* can claim that "Classified Class 0 is for sure class 0." On the other hand, the cluster associated with class "0" (contingency table (b)) has implicated correct class 8 times and 2 times an incorrect class. Therefore, based on the contingency table (b) we can state that the output is a class "*0"* more likely than class *"1"*, but for sure, it is an NOT class *"2."* Such statements in the human-friendly form may support a decision making.

| **Contingency table (a)** | | **Predicted Class** |
|---|---|---|
| | | *Class = 0* |
| **Desired output class** | *Class = 0* | *10* |
| | *Class = 1* | *0* |
| | *Class = 2* | *0* |

| **Contingency table (b)** | | **Predicted Class** |
|---|---|---|
| | | *Class = 0* |
| **Desired output class** | *Class = 0* | *8* |
| | *Class = 1* | *2* |
| | *Class = 2* | *0* |

Tab 6. Examples of contingency tables for two different clusters

## V. Conclusion

The main goal of this study is to provide scientific evidence behind the claimed characteristics of MF ARTMAP neural network and improve the accuracy assessment the network. We empirically showed that MF ARTMAP could be trained in incremental nature due to its self-organization capacity and establishment of new clusters.

Further results of this paper are to improve generalization and also to improve accuracy as well, which was achieved by a Simulated Annealing approach and resulted in obtaining the best ratio between network generalization and classification precision by optimization parameters of the fuzzy clusters.

Performed experiments showed that parameters optimization dramatically increases the classification accuracy of the MF ARTMAP network. In comparison with the MLP, we can conclude, that MF ARTMAP's classification is still slightly worse, but its incremental learning capability may outperform many others neural networks, including MLP.

Furthermore we have argued about the potential of the MF ARTMAP algorithm towards to processing information about classified classes and their structure in the feature space in the intra-class and inter-class manner.

Since the MF ARTMAP is incremental algorithm with good learning plasticity, we see a potential in deployment in the cloud environment, where we can benefit from crowdsourced collective learning, sharing gained knowledge between masses persons and devices and at last but not least development of multi-agent systems. The realization of cloud-based MF ARTMAP is our immediate future target.


## Acknowledgement

This research is supported by the National Research and Development Project Grant 1/0773/16 2016 – 2019 "Cloud Based Artificial Intelligence for Intelligent Robotics" and by the Slovak Research and Development Agency under the contract No. APVV-015-0731 and research project is supported from 07-2016 to 06-2019.


## Bibliography


[1] W. McCulloch and W. Pitts, "A logical calculus of the ideas immanent in nervous activity," *Bull. Math. Biophys.*, vol. 5, no. 4, pp. 115–133, 1943.

[2] R. M. Cesar and L. da Fontoura Costa, "An introduction to neural networks," *Neurocomputing*, vol. 14, no. 1, p. 104, 1997.

[3] M. McCloskey and N. J. Cohen, "Catastrophic interference in connectionist networks: The sequential learning problem," in *Psychology of Learning and Motivation*, vol. 24, G. H. Bower, Ed. San Diego, CA: Academic Press, 1989, pp. 109–164.

[4] S. Grossberg, "How does a brain build a cognitive code?," *Psychol. Rev.*, vol. 87, no. 1, pp. 1–51, 1980.

[5] G. Carpenter, "Adaptive Resonance Theory," (tech. report) Boston, 2000.

[6] P. Smolár, P. Sinčák, and M. Virčíková, *Object Categorization with Artmap Neural Networks*, vol. 1. Saarbrucken: LAP LAMBERT Academic Publishing, 2012.

[7] G. Carpenter and S. Grossberg, "ART 2: self-organization of stable category recognition codes for analog input patterns.," *Appl. Opt.*, vol. 26, no. 23, pp. 4919–4930, 1987.

[8] G.. Carpenter, S. Grossberg, and D. Rosen, "ART 2-A: an adaptive resonance algorithm for rapid category learning and recognition," *IJCNN-91-Seattle Int. Jt. Conf. Neural Networks*, vol. ii, no. 4, pp. 493–504, 1991.

[9] G. Carpenter, S. Grossberg, and D. Rosen, "Fuzzy ART: Fast, stable learning and categorization of analog patterns by an adaptive resonance system," *Neural Networks*, vol. 4, no. 6, pp. 759–771, 1991.

[10] P. Sinčák, M. Hric, and J. Vaščák, "Membership Function-ARTMAP Neural Networks," *TASK Q.*, vol. 7, no. 1, pp. 43–52, 2003.

[11] P. Sinčák, M. Hric, and J. Vaščák, "Neural Network Classifiers based on Membership Function ARTMAP," in the *Systematic organization of information in fuzzy systems*, 2003, pp. 321–333.

[12] A. Bodnárová, "The MF-ARTMAP neural network," *Latest Trends Appl. informatics Comput.*, pp. 264–269, 2012.

[13] M. Lichman, "UCI Machine Learning Repository," *UCI Machine Learning Repository*, 2013. [Online]. Available: https://goo.gl/BWctV6. [Accessed: 20-Jun-2016].

[14] J. Stefanowski, "Data Mining - Evaluation of Classifiers," *Lecture*. Institute of Computer Sciences, Poznan University of Technology, Poznan, p. 36, 2010.

[15] E. Aarts and J. Korst, *Simulated annealing and boltzmann machines*. New York: John Wiley and Sons Inc., 1988.

[16] R. Landis and G. Koch, "The Measurement of Observer Agreement for Categorical Data," in *Biometrics*, 1977, pp. 159–174.

[17] N/A, "Confusion Matrix–Another Single Value Metric–Kappa Statistic," 2011. [Online]. Available: http://goo.gl/XTujN. [Accessed: 20-Jun-2016].