WeVoS-ViSOM: An Ensemble Summarization Algorithm for Enhanced Data Visualization

Emilio Corchado[1] and Bruno Baruque[2]

1.- Department of Informatics and Authomation. UNIVERSITY of SALAMANCA

2.- Department of Civil Engineering. UNIVERSITY of BURGOS.

Escuela Politécnica Superior - Campus Vena (Edif.C). C/ Francisco de Vitoria, s/n. 09006 Burgos, SPAIN.

Tel: (+34) 974259513; Fax: (+34) 947 259000

Abstract

This study presents a novel version of the Visualization Induced Self-Organizing Map based on the application of a new fusion algorithm for summarizing the results of an ensemble of topology preserving mapping models. This algorithm is referred to as Weighted Voting Superposition (WeVoS). Its main feature is the preservation of the topology of the map, in order to obtain the most truthful visualization of datasets under study as possible. To achieve this, a weighted voting process takes place between the units of the maps in the ensemble in order to determine the characteristics of the units of the resulting map. In order to present a thorough study of its capabilities, several different quality measures have been applied and analysed under this novel neural architecture called WeVoS-ViSOM. To complete the study,it has also been compared with with the well-know SOM and its fusion version, the WeVoS-SOM and with two other previously devised fusion algorithms - Fusion by Euclidean Distance and Fusion by Voronoi Polygon Similarity - based on the analysis of the previous same quality measures in order to present a thorough study of its capabilities. All three summarization methods were applied to three widely used datasets from the UCI Repository and after a rigorous performance analysis, it is clearly demonstrated that the novel fusion algorithm outperformed the other single and summarization methods in terms of visualization of the datasets.

Keywords: Topology preserving maps, unsupervised learning, data visualization, ensembles, summarization algorithm.

Email address: escorchado@usal.es; bbaruque@ubu.es (Emilio Corchado[1] and Bruno Baruque[2])

1. Introduction

Among the great variety of visualization tools for multidimensional datasets, one of the most well-known are the Topology Preserving Maps. The ViSOM Yin [34, 33] is a very interesting extension of the well-known Self-Organizing Map (SOM) Kohonen [18], Kohonen et al. [21], Kohonen [19] characterized by being capable of representing quantitatively the similarity between the data it is analysing.

This family of models allow the representation of high-dimensional datasets into 2-dimensional maps and facilitate to the human expert the interpretation of the internal structure of data. They are also characterized by the use of unsupervised and competitive learning.

The main problem of all the neural network algorithms in general is that, they are rather unstable Heskes [13], Bakker and Heskes [3]. Running the same algorithm, even using the same parameters; can lead to quite different results. The use of ensembles is one of the most spread techniques for increasing the stability of an analysis model Schwenk and Bengio [29], Johansson et al. [15]. This meta-algorithm consists in training several slightly different models over the same data set and relying on their combined results, rather than in the results of a single model. This is based in the intuitive idea that a committee of experts working to solve a particular problem would come up with a more reliable solution than a single expert working in the same problem.

This technique is used in a great number of studies, applied mainly to classification problems. In this study, however the desired result is to obtain the most reliable as possible representation of a multidimensional data set on a 2-dimensional map. Therefore, the classical ensemble summarization techniques are not directly applicable in this case.

Several algorithms for topographic maps summarization have been previously proposed Petrakieva and Fyfe [23], Georgakis et al. [10], Saavedra et al. [28], although there are some characteristics of the topology preserving models that have not been taken into account. In this research it is presented and analysed a novel fusion version of the ViSOM called the WeVoS-ViSOM and it is compared with the single SOM and ViSOM and its WeVoS fusion version. The study reports the application of these algorithms to three of the most widely-known datasets of the UCI web repository Asuncion and Newman [1]: Iris, Wine and Wisconsin Breast Cancer.

The rest of this study is organized as follows: Section 2 introduces the Topology preserving mapping. Section 3 presents five quality measures, previously proposed in literature, used to evaluate different properties of topology preserving mapping algorithms in general. Section 4 includes a brief description of the ensemble meta-algorithms and several previously proposed algorithms for summarizing SOM ensembles. Section 5 describes in detail the novel proposed summarization method: the Weighted Voting Superposition ViSOM (WeVoS-ViSOM). Section 6 describes the evaluation of the properties of the summaries obtained by the WeVoS-ViSOM algorithm and compares them with those calculated for the maps generated by the single models and other summarization methods. Finally, in Section 7 final conclusions and future lines of research are outlined.

2. Topology Preserving Maps

Topology preserving maps Kohonen et al. [21], Kohonen [20] comprises a family of techniques with the target of producing a low dimensional representation of the training samples while preserving the topological properties of the input space. The best known technique among them is the Self-Organizing Map (SOM) model Kohonen [19]. It is based on a type of unsupervised learning called competitive learning; an adaptive process in which the units in a neural network gradually become sensitive to different input categories, sets of samples in a specific domain of the input space. The main feature of the SOM algorithm is its topology preservation. When not only the winning unit but also its neighbours on the lattice are allowed to learn, neighbouring units gradually specialize to represent similar inputs, and the representations become ordered on the map lattice.

One interesting extension of this algorithm is the Visualization Induced SOM (ViSOM) Yin [34], Gou et al. [11], proposed to directly preserve the local distance information on the map, along with the topology. The ViSOM constrains the lateral contraction forces between units and hence regularises the inter-unit distances so that distances between units in the data space are in proportion to those in the input space. The ViSOM does not only take into account the distance between a unit's weights and the input data entry for the update of a unit's weights, but also the distance between that unit and the best matching unit of the whole map.

The difference between the SOM and the ViSOM hence lies in the update of the weights of the neighbours of the winner unit as can be seen from Eq. 1 and Eq. 2. Update of neighbourhood units in SOM:

$$w_k(t+1) = w_k(t) + \alpha(t)\eta(v,k,t)(x(t) - w_k(t)) \tag{1}$$

Update of neighbourhood units in ViSOM:

$$w_k(t+1) = w_k(t) + \alpha(t)\eta(v,k,t)\left[(x(t) - w_v(t)) + (w_v(t) - w_k(t))\frac{d_{vk} - \triangle_{vk}\lambda}{\triangle_{vk}\lambda}\right] \tag{2}$$

where x is the input to the network, w_k is the weight vector associated with neuron k, while w_v is the weight vector associated to the winning unit in the lattice, also called Best Matching Unit (BMU), $\alpha(t)$ is the learning rate of the algorithm; $\eta(v,k,t)$ is the neighbourhood function (usually a Gaussian function), where v represents the position of the BMU for the particular x of time t and k the positions of the units in the neighbourhood of this one. λ is a "resolution" parameter, d_{vk} and $\triangle_{vk}$ are the distances between the units in the data space and in the map space respectively.

3. Features to Analyse

Several quality measures have been proposed in literature to study the reliability of the results displayed by topology preserving models in representing the data set that have been trained with Polani [24], Pozlbauer [25]. There is not a global and unified one, but rather a set of complementary ones, as each of them asses a specific characteristic of the performance of the map in different visual representation areas. Five of them are briefly presented in the following section. These measures have been chosen with the objective of measuring as a wide range of these characteristics as possible.

As stated in the introduction, the aim of the novel model presented (WeVoS-ViSOM) is to obtain a truthful representation of the data set in a map to obtain the best possible visualization of the internal structure of a data set. Thus, the most important features to evaluate in this case are the neighbouring relationship of the units of the map and the continuity of the map. These features are assessed by topographic error, distortion and to some extent goodness of map. The two remaining measures, (classification accuracy and mean square quantization error) complete the comparison of the models in this research.

Topographic Error . Kiviluoto [17] is calculated by finding the first two best matching units for each entry of the data set and testing whether the second is in the direct neighbourhood of the first or not. This measure, although suitable for an approximation of the quality of a map, is considered somehow simplistic and therefore not completely reliable in some cases by several studies Pozlbauer [25].

Distortion . Vesanto et al. [31] : When using a constant radius for the neighbourhood function of the learning phase of a SOM; the algorithm optimizes a particular function. This function can be used to quantify in a more trustful way than the previous one the overall topology preservation of a map by means of a measure called distortion measure in this work. Special attention is paid to this measure in this research due to its relation with visualization properties.

Classification Accuracy . Topology preserving models can be easily adapted for classification of new samples using a semi-supervised procedure Vesanto [30]. Once the network training is completed, the same data set used in the training stage is presented once again to the network. Each unit of the map is labeled with the class it has most consistently recognized. When a new sample is presented to the network, it is classified by the class associated to the unit that is activated at that time. A high value in the classification accuracy rate implies that the units of the map are reacting in a more consistent way to the classes of the samples that are presented. As a consequence, the map should represent the data distribution more precisely.

Mean Square Quantization Error . can be calculated for any algorithm performing vector quantization. In this case, it indicates how well the units of the map approximate the data on the data set. Or in other words, it measures the closeness of the units composing the map to the different data entries they recognize (i.e., are considered as the BMU for that entry); in the input space.

Goodness of Map . Kaski and Lagus [16] combines two of the previous error measures: the square quantization error and the distortion. It takes account of both the distance between the input and the BMU and the distance between the first BMU and the second BMU in the shortest path between both along the grid map units, calculated solely with units that are direct neighbours in the map. Thus, it measures both the continuity of the mapping from the data set to the map grid and the accuracy of the map in representing the set.

4. Topology Preserving Mapping Fusion

4.1. Use of Ensemble Meta-Algorithms

The use of an ensemble of similarly trained models or algorithms is intended to improve the performance of classification algorithms Breiman [7]. It has been observed that, although one of the classifiers in an ensemble would yield the best performance, the sets of patterns misclassified by the different classifiers would not necessarily overlap. As a conclusion, different classifier designs potentially offer complementary information about the patterns to be classified and could be harnessed to improve the performance of the selected classifier. The aim is not to rely on a single decision making scheme, but rather use all the designs or their subsets for decision making, by combining their individual opinions to derive a consensus decision Ruta and Gabrys [27], Henriques et al. [12].

The main problem of competitive-learning-based networks is that are inherently unstable due to the random nature of their learning algorithm. The leading idea of this research is that the effect of this instability may, however, be minimized by the use of ensembles Ron and Gunnar [26], Wang et al. [32]. The learning algorithm of the topology preserving maps family trains their composing units (or neurons) to specialize during the algorithm iterations in recognizing a certain type of patterns, which determines also the topology of the map. In a similar way to the classification process, it can be inferred that the map regions that do not accurately represent the nature of the data set do not necessarily overlap. Therefore, the visualization of a single map might be improved by adapting each of the composing units of a map in the best possible way to the data set under study by using ensemble techniques, as they offer complementary visualizations of the data set.

Algorithms to combine classifiers can be divided into two broad classes. The simpler variety of algorithms Breiman [7] merely combines, by averaging in some way, the results of each of the classifiers into a final result. More complex types of algorithms Dietterich [9], Kuncheva [22] try to combine not only the results, but the whole set of classifiers; in an attempt to construct a single one that should outperform its individual components. Its main advantage is that it combines an improvement on the classification quality with the simplicity of handling only one classifier.

This perspective of a single "summary" or "synthesis" of the patterns stored within the whole ensemble is the one followed in the present research to improve the model performance. The main intention is to obtain a unique map capable of representing the different features contained in the different maps of the ensemble in the clearest and most reliable way as possible.

4.2. Summary of the Topology Preserving Map Ensemble

The models used in this study are mainly designed as visualization tools. Constructing ensembles of classifier models is a viable option when trying to boost their classification capabilities, stabilizing its learning algorithm and avoiding overfitting; but when dealing with its visualization feature an ensemble is not directly displayable. Representing all the maps in a simple image can be useful when dealing with only 1-dimensional maps Petrakieva and Fyfe [23], but is unmanageable when visualizing 2-D maps. As a part of this research, a novel ensemble combination algorithm has been devised to overcome this problem, by generating a unique map representing the information contained in the different maps composing the ensemble. This combination algorithm is intended to generate an accurate and stable representation of data for visual inspection.

This part of the study encompasses several approaches inspired by previously developed work regarding SOM combination Baruque et al. [5, 6]. The study also includes previously presented methods centred on the generation of a final map summarizing the contents of several maps Georgakis et al. [10], Saavedra et al. [28] for comparison purposes. Hereafter this process is called "Fusion". The main characteristics of two of those methods are briefly described in Section 4.2.1 and their performance results are showed and discussed in Section 6.

Then, a novel approach to the fusion of maps is presented in this work (WeVoS-ViSOM). It is fully described in Section 5 and its performance is compared with previous devised algorithms in Section 6.

4.2.1. Previous Work: Fusion of SOMs

In this study the presented model is compared with the two fusion algorithms previously presented for this same pourpose known by authors. Although these algorithms have been developed by different authors and for different tasks, both employ a similar approach to this task; but different from the one introduced in this work. Therefore, in this case both previous approaches are considered two variants of the same 'parent' algorithm, while the WeVoS-ViSOM is considered a different one.

The previous Fusion of SOM meta-algorithm involve comparing the maps unit by unit in the input space. That is, units that are considered 'near enough' one to the other are fused to obtain a unit in the final fused map. This is done by calculating the centroid of the weights of the units to fuse:

$$w_c = \frac{1}{|W_k|} \sum_{w_i \in W_k} w_i \tag{3}$$

being W_k the characteristic vectors of the set of units to fuse. That process is repeated until all units in all trained maps are fused into a unique final one. The criteria to determine which units are 'near enough' to be fused is what determines the two variants of the main algorithm.

Criterion 1: Voronoi Polygons Saavedra et al. [28]. . Each unit in a Self-Organizing Map can be associated with a portion of the input data space called the Voronoi polygon Aurenhammer and Klein [2]. That portion of the multi-dimensional input space contains data for which that precise unit is the BMU of the whole map. It is therefore logical to conclude that units related to similar Voronoi polygons can be considered similar between them, as they should be situated relatively nearby in the input data space.

To calculate the dissimilarity between two units, a record of which data entries activated each unit as the BMU can be stored by associating a binary vector to the unit which length is the size of the data set. The vector will contain ones (1) in the positions where the unit was the BMU for that sample and zeros (0) in the rest of positions. The dissimilarity (i.e. the distance) between units can therefore be calculated as in Eq. 4:

$$ds(b_r, b_q) = \frac{\sum XOR(b_r, b_q)}{\sum OR(b_r, b_q)} \tag{4}$$

being r and q the units to determine their dissimilarity and b_r and b_q the binary vectors relating each of the units with the data samples recognized by it.

The main problem with this proximity criterion is that it depends on the recognition of data by the map, rather than on the map itself. This means that a unit that does not react as the BMU for any data could be considered similar to another unit in the same condition, although they can be relatively far from each other in the input data space. To overcome this problem, all units with a reacting rate lower than a threshold are removed before calculating the similarities between the remaining units. This implies that the neighbouring properties of the whole map are no longer considered. The similarity criteria must be used again to keep a notion of neighbouring between the units of the fused map. Units whose dissimilarity is below a given threshold will be considered as neighbours in the fused map.

This characteristic can be very useful when the objective of the analysis is to learn and represent the topology of the data set, as the remaining units will approximate the data set in the input space very well, enhancing the vector quantization feature of the SOM. Its drawback is that it is not possible to represent that structure in a 2D map, as a lot of neighbouring information between units is disregarded. The process is fully described in Algorithm 1.

Criterion 2: Euclidean Distance Georgakis et al. [10]. . This method involves comparing the maps unit by unit in the input space, which implies that all the maps in the ensemble must have the same size. Firstly, it searches for the units that are closer in the input space (selecting only one unit in each map of the ensemble) then it "fuses" them to obtain the final unit in the "fused" map

Algorithm 1 Map Fusion by Voronoi Polygon Similarity

Input: Set of trained topology-preserving maps: $M_1 ... M_n$,
usage threshold: θ_u , fusion threshold: θ_f, connection threshold: θ_c
Output: A final fused map: M_{fus}

Select a training set $S = \langle (x_1, y_1) \ldots (x_m, y_m) \rangle$
train several networks by using the bagging (re-sampling with replacement) meta-algorithm : M_n
let θ_u, θ_f and θ_c be the usage, fusion and connection thresholds respectively
procedure FUSION($M_1 ... M_n$)
    for all $M_i \in M_n$ **do** ▷ for all maps in the ensemble
        for all $w_j \in W_i$ **do** ▷ for all neurons in each map
            ▷ accept neurons with a recognition rate higher than a given threshold
            $W_{fus} \leftarrow w_i \ if \ \sum_i b_r(i) > \theta_u$
        end for
    end for
    for all $w_i \in W_{fus}$ **do**
        calculate dissimilarity between w_i and ALL neurons in W_{fus} (Eq. 4)
        $D_i \leftarrow ds(w_i, w_k) \forall w_k \in W_{fus}$
    end for
    group into different sub-sets (Ws_n) the neurons that satisfy the conditions of $\begin{cases} ds(b_r, b_q) < \theta_f & \forall r, q \in Ws_n \\ ds(b_r, b_q) > \theta_f & \forall r, q \notin Ws_n \end{cases}$
    for all Ws_n **do**
        calculate the centroid (w_c) of the set by using Eq. 3
        add the centroid to the set of nodes of the final map (W^*_{fus})
    end for
    for all $w_r \in W^*_{fus}$ **do** ▷ for all neurons in the fused map
        Connect w_r with any other neuron in W^*_{fus} , if they satisfy $\min_{b_r \in Ws_k, b_q \in Ws_l} ds(b_r, b_q) < \theta_c$
    end for
end procedure

(see Eq. 3). This process is repeated until all the remaining units have been fused. The high computational complexity of the algorithm is approached by using dynamic programming. The final fused map is initialized by calculating the fusion of only two of the maps composing the ensemble. Then, the same calculation is repeated between the resultant fused map and another one of the maps composing the ensemble. The process continues until all the maps of the ensemble have been included in the calculation of the fused map.

The difference with the previous criteria is that, in this case, a pair wise match of the units of each map is always possible, so the final fused map has the same size as each of its constituent ones. This also implies that a certain global neighbouring structure can be maintained and reconstructed in the fused network. The algorithm that employs this criterion is fully described in Algorithm 2.

Algorithm 2 Map Fusion by Euclidean Distance

Input: Set of trained topology-preserving maps: $M_1 ... M_n$
Output: A final fused map: M_{fus}

Select a training set $S = \langle (x_1, y_1) \ldots (x_m, y_m) \rangle$
train several networks by using the bagging (re-sampling with replacement) meta-algorithm : M_n
procedure FUSION(M_n)
    initialise M_{fus} with the weight vectors of the first map: $M_{fus} \leftarrow M_1$
    for all $M \in M_n$ **do**
        for all $w'_i \in M_{fus}$ **do**
            calculate Eucl. Dist. between w'_i and ALL neurons of map M_i
            ▷ let w^* be the closest neuron in map M_i to the one selected in M_{fus}
            $w^* \leftarrow argmin_i \left(ED(w'_i, w_i) \right)$
            $w_c \leftarrow w'_i + w^*/2$ ▷ applying Eq. 3 to two neurons
            $w'_i \leftarrow w_c$ ▷ replace w_i by the centroid (w_c)
        end for
    end for
end procedure

5. Weighted Voting Superposition for ViSOM

The idea behind this novel fusion variant presented in this study –WeVoS-ViSOM– is to obtain the final map in a unit by unit basis. However, instead of aiming for the best position for a single unit, as the two previously explained methods, this approach aims to obtain the best position for a unit and their neighbours. As a consequence, the final map obtained keeps one of the most important features of this type of algorithms: its topological ordering. This is an interesting characteristic having into account that the principal characteristic of the ViSOM is the enhance of the data visualization. Also, the modified weights update of the ViSOM, which provides the units with more freedom to adapt to the data set, potentially adds instability to the training. The WeVoS-ViSOM, because of its acknowledge of the neighbouring of units in its process, seems to be the most suitable fusion algorithm to diminish this effect.

The WeVoS scheme is an improved version of an algorithm presented in several previous works: superposition Corchado et al. [8]. It has been applied to the simple SOM in previous works Baruque and Corchado [4] with interesting results.

The first step in this meta-algorithm is to calculate the "quality" of each of the units composing each map, in order to rely on some kind of informed decision for the fusion of neurons. This "quality" (or error) measure can be any of the many quality of map measures existing in literature on Topology Preserving Maps; provided that it may be calculated on a unit-by-unit basis.

The final map is obtained again in a unit by unit basis. Firstly, the units of the final map are initialized by calculating the centroids of the units in the same position of the map grid in each of the trained maps. Then, a recalculation of the final vector of that unit is done using the information associated to the units in that same position in each map of the ensemble. For each unit, a sort of voting process is performed as seeing in Eq. 5:

$$V_{p,m} = \frac{\sum b_{p,m}}{\sum_{i=1}^{M} b_{p,i}} \cdot \frac{\sum q_{p,m}}{\sum_{i=1}^{M} q_{p,i}} \tag{5}$$

where, $V_{p,m}$ is the weight of the vote for the unit included in map m of the ensemble, in its in position p, M is the total number of maps in the ensemble, $b_{p,m}$ is the binary vector used for marking the dataset entries recognized by unit in position p of map m, and $q_{p,m}$ is the value of the desired quality measure for unit in position p of map m. The weights of the units are fed to the final network as it is done with data inputs during the training phase of a topology preserving map, considering the homologous unit in the final map as the BMU. The weights of the final unit will be updated towards the weights of the composing unit. The difference of the updating performed for each homologous unit in the composing maps depends on the quality measure calculated for each unit: the higher the quality (or the lower the error) of the unit of the composing map, the stronger the updating of the unit of the summary map towards the weights of that particular unit. A single measure or a linear combination of several quality measures can be used for the determination of the final quality of a unit. The number of data inputs recognized by each unit is also taken into account in the quantization of the "most suitable" unit among those voting for the same position in the final map.

In short, the summarization algorithm considers the weights of a composing unit "more suitable" to be the weights of the unit in the final map according to both the number of inputs recognized and the quality of adaptation of the unit. The steps of this algorithm are fully described in the Algorithm 3.

Its interesting to note that the WeVoS scheme leans in the base algorithm used to update the final units and its neighbourhood. That means that although very similar, the WeVoS-SOM is different form the WeVoS-ViSOM algorithm.

Algorithm 3 Weighted Voting Summarization algorithm

Input: Set of trained topology-preserving maps: $M_1 ... M_n$, training data set: S
Output: A final fused map: M_{fus}

Select a training set $S = \langle (x_1, y_1) \dots (x_m, y_m) \rangle$
train several networks by using the bagging meta-algorithm : $M_1 ... M_n$
procedure WEVOS-VISOM($M_1 ... M_n$)
  for all map $M_i \in M_n$ **do**
    calculate the quality/error measure chosen for ALL neurons in the map
  end for
  ▷ These two values are used in Eq. 5
  calculate an accumulated total of the quality/error for each position $Q(p)$
  calculate recognition rate for each position $B(p)$.
  for all unit position p in M_i **do**
    initialize the fused map (M_{fus}) by calculating the centroid (w_c) of the neurons of all maps in that position (p) Eq. 3
  end for
  for all map $M_i \in M_n$ **do**
    for all unit position p in M_i **do**
      calculate the vote weight (V_{p,M_i}) using Eq. 5.
      feed the weights vector of neuron w_p into the fused map (M_{fus}) as if it was an input to the network.
      The weight of the vote (V_{p,M_i}) is used as the learning rate (α).
      The position of that neuron (p) is considered as the position of the BMU (v). ▷ This causes the neuron of the fused map (w_p^*) to approximate the neuron of the composing ensemble ($w_{p,m}$) according to the quality of its adaptation.
    end for
  end for
end procedure

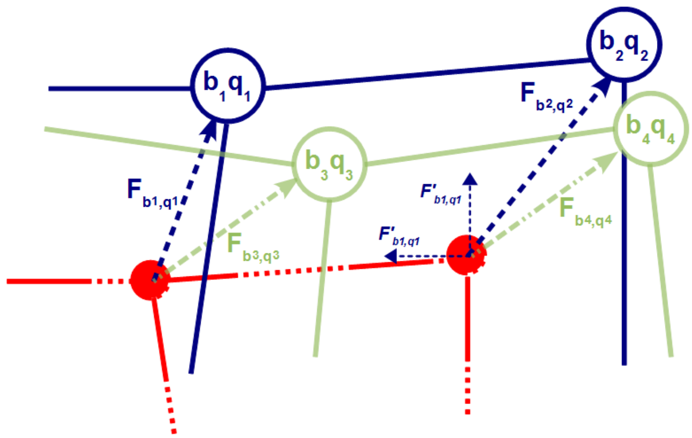


Figure 1: Final adaptation of units in WeVoS-ViSOM

When the step of "fine tunning" of units according to composing maps is performed, the WeVoS-SOM uses the SOM updating (Eq. 1) while the WeVoS-ViSOM uses the ViSOM updating (Eq. 2).

An schema representing how the final update of units in the WeVoS-ViSOM is performed is shown in Fig. 1. As can be seen, when the update of the final unit on the left is performed according to the homologous unit of the first map, its neighbour (final unit on its right) is also updated according to its distance with the unit being updated (in this case the one in the right) contracting (or expanding) the grid as is done in the ViSOM algorithm. This is repeated for all units of each composing map. This difference can be empirically appreciated in the experiments presented in Section 6.

6. Experiments and Results

Several experiments have been designed and performed to investigate the capabilities of the WeVOS-ViSOM and also to compare it versus the other two different algorithms for obtaining a fused map from an ensemble. These experiments made use of three of the most popular datasets included in the UCI machine learning repository Asuncion and Newman [1]: Iris, Echo-Cardiogram and Wine datasets. Experiments were performed using both ViSOM and SOM models over the three datasets to train ensemble of different sizes, using the classical cross-validation method in order to select testing and training parts of the corresponding dataset.

6.1. Test Procedure

For all the experiments involving this combination of maps the procedure is the following: A simple n-fold cross-validation is used in order to employ all data available for training and testing the model and to calculate an average of its performance. An ensemble of maps is calculated in each step of the cross-validation. The way the ensemble is trained does not affect the way the combination is computed. In the case of this study this has been done using the bagging Breiman [7] meta-algorithm. Each individual map of an ensemble is trained on one of the re-sampled subsets ($n-1$ folds of the whole dataset) initialized in the same way and using exactly the same parameters for training. This generates n different trained networks which can be combined into a final network that is expected to outperform each of them individually. The combination of maps is done once all the maps composing the ensemble have finished their training. Then, the data fold that was left out of the training resampling, is used to test all models trained: each of the networks that compose the ensemble as well as the combinations that they generate.

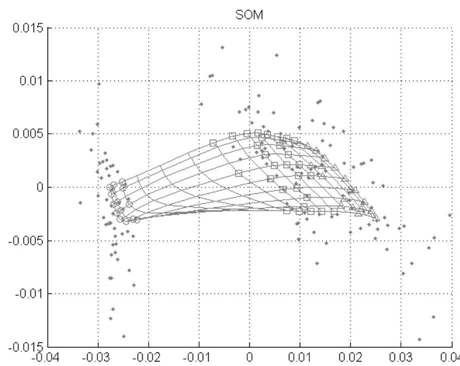
6.2. Visualization Results

In this sub-section a few examples of the most interesting visualization results obtained by the different models discussed in this research are presented.

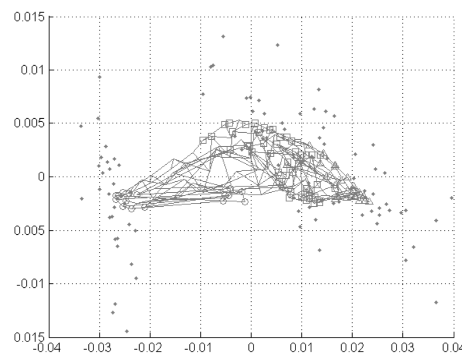
Figs. 2 and 3 represent how each map adapts its structure to represent the data set analyzed. It depicts the lattices composing the maps embedded in a 2-dimensional input space. All figures represent the Iris dataset projected over its first 2 principal components Hotelling [14]. X-axis is the 1^{st} PC and Y-axis is the 2^{nd} PC. Each of the figures shows also a map trained over the dataset, embedded into the input space formed by the principal components. As before, all maps were training using the same parameters.

In Figs. 2a and 3a it can be seen the single model maps displayed over the Iris dataset in the input space. The first displays a SOM grid, the later a ViSOM grid. In Figs. 2b and 3b the result of performing the algorithm of the Fusion by Euclidean distance is showed. It is easy to observe that this algorithm is focused on distributing the units over the dataset the best as possible, but obtains a map with a lot of twists and folds that does not preserve the topology very well. In Figs. 2c and 3c we display the fusion of the same ensemble, but using the fusion by Voronoi polygon similarity. Again, only the map training algorithm changes in the two figures. It can be observed that the topology preservation is completely lost. Finally, Figs. 2d and 3d show the fusion of the ensemble, using the model presented in this work: the WeVoS (WeVoS-SOM in the first, WeVoS-ViSOM in the second figure). It can be seen that the problems observed previously are not present in this model.

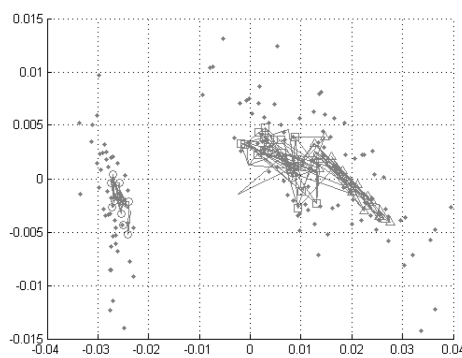
The resultant maps that will be used to visually inspect the data obtained for the single model and two of the different summarization models described in this study are displayed on Fig. 4 for both the SOM and ViSOM map training algorithms. As explained before, the Voronoi Polygon Similarity Fusion algorithm does not form a proper lattice, but rather a graph; and therefore it is not suitable for data representation as the rest of them.



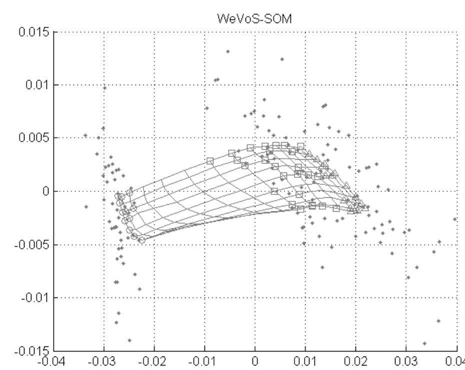
(a) Single SOM map



(b) Fusion of SOM maps using the Euclidean Distance criterion

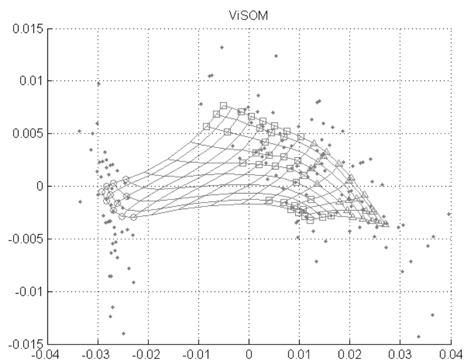


(c) Fusion of SOM maps using the Voronoi polygon similarity criterion

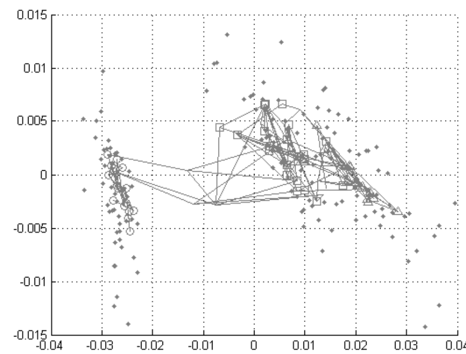


(d) Fusion of five SOM maps using the WeVoS algorithm

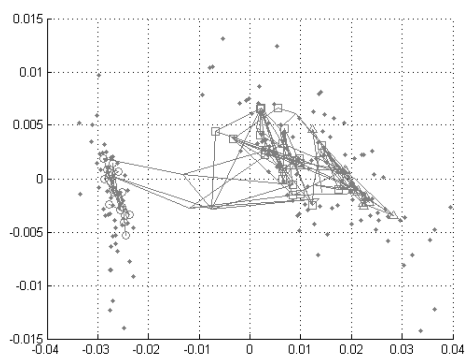
Figure 2: Comparison of the adaptation the discussed algorithms to input data set using the SOM as base component for Iris data set.



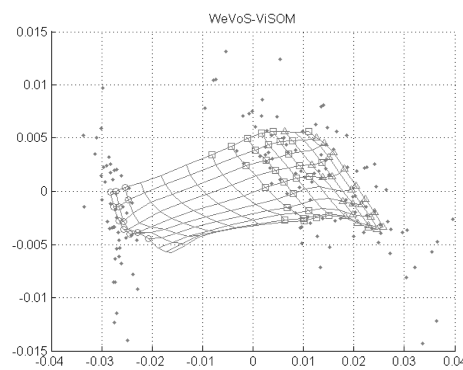
(a) Single ViSOM map



(b) Fusion of ViSOM maps using the Euclidean Distance criterion



(c) Fusion of ViSOM maps using the Voronoi Polygon Similarity criterion



(d) Fusion of ViSOM maps using the WeVoS algorithm

Figure 3: Comparison of the adaptation the discussed algorithms to input data set using the ViSOM as base component for Iris data set.

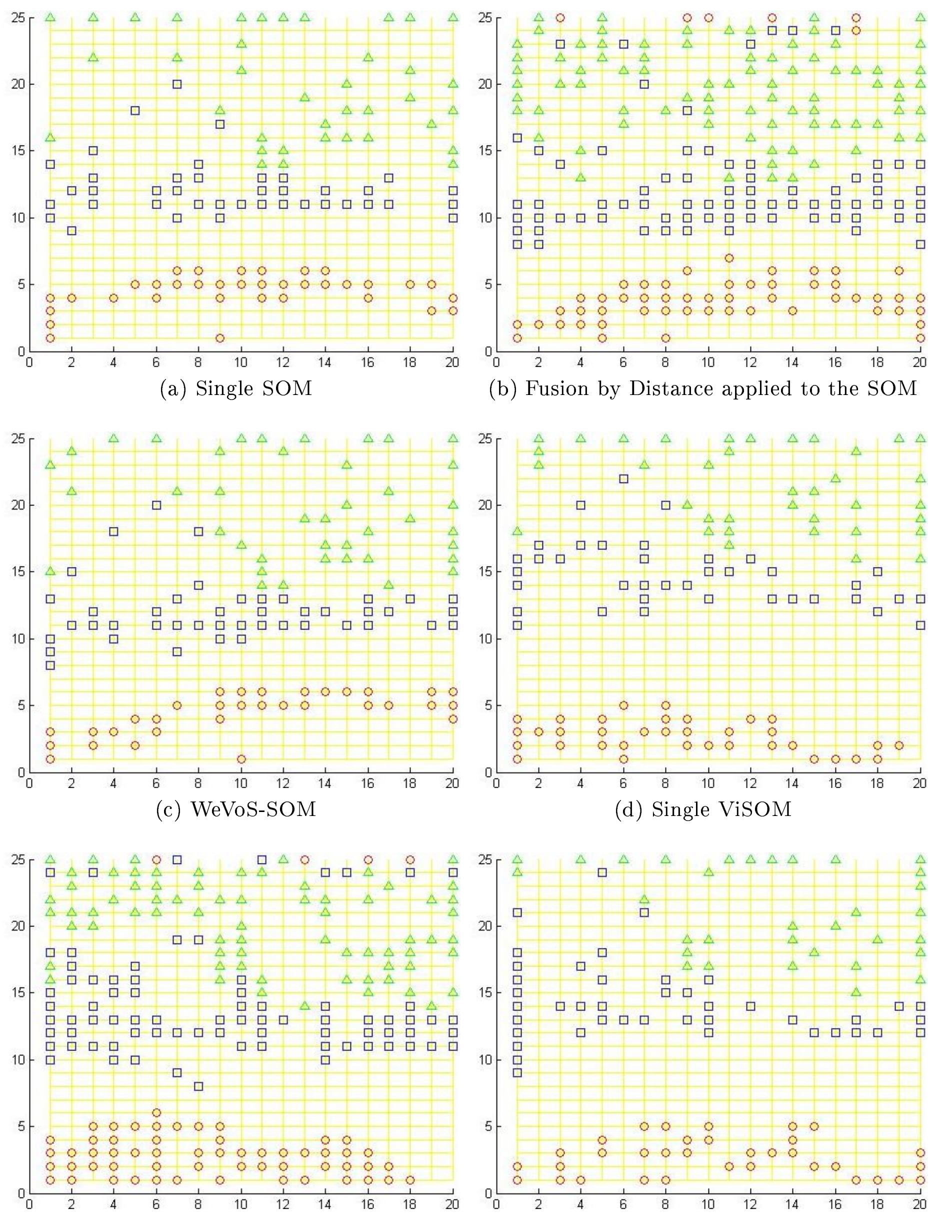
The six maps were trained using the iris data set. Ensemble fusion maps were obtained from the summarization of the same 7 maps (except Figs. 4a and 4d that show the single map SOM and ViSOM versions). Each unit of the map is represented according to the class it has recognized more consistently.

It can be seen from Fig. 4 that the ViSOM algorithm provides in general a smoother map than the classic SOM algorithm. The WeVoS meta-algorithm improves the Single and Fusion by Distance models by obtaining in general more compact and more clearly separated groups than the other two. Compared with the single model (Fig. 4a), the WeVoS-SOM (Fig. 4c) presents a much separated group for class 1 (circles). While in the single map, class 1 appears in a strip on the left corner on the map, leaving a considerable amount of dead units between data and the border of the map; the WeVoS-SOM presents a more separated group covering the top of the map. The Fusion by Distance (Fig. 4b) summarization algorithm does not improve significantly the single map, regarding the data representation feature of the model, as it obtains a map mixing the three classes in the top part of the map. The WeVoS-ViSOM provides a better visualization compared with the single ViSOM model and the other two summarization algorithms. The single ViSOM (Fig. 4d) represents the iris dataset quite well, with a group of samples corresponding to class 1 clearly separated from the others. Although they also appear in a corner of the map, the cluster of class 1 is more separated from the other cluster than in the classic SOM model (Fig. 4a). The Fusion by Distance (Fig. 4e) map also contains a group separated from the rest, but including samples of different classes. As explained before, the Fusion by Similarity is not suitable for 2-dimensional map representation, as some units are disregarded from the final model and therefore the topology preservation is lost. The WeVoS-ViSOM (Fig. 4f) clearly separates class 1 from the other two in a more compact group in the top of the image. Even comparing this model with the rest of models presented in Fig. 4, the WeVoS-ViSOM is the one that separates more clearly class 1 from the rest. The other two classes, although not so clearly separated as the first one, also appear unmixed between them and in a more compact group than in the single ViSOM map (Fig. 4d).

6.3. Analytical Results

This sub-section includes complete results for the experiments performed comparing the models previously discussed according to the analytical quality measures presented in Section 3. Two different sets of experiments were performed to compare the performance of the models when varying two different aspects of the training.

All measures presented in this section, are error measures; so the desired value is always as close to 0 as possible. The Classification Error is presented in percentage form, normalized between 0 and 1, while the rest of the measures are absolute values. For the sake of clarity, the results for the Fusion by Voronoi Polygon Distances have been let out of the comparative, as they are completely different from the rest and therefore, are not comparable.



(a) Single SOM

(b) Fusion by Distance applied to the SOM

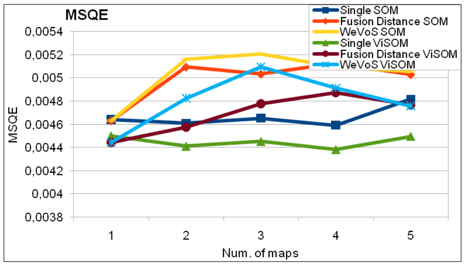
(c) WeVoS-SOM

(d) Single ViSOM

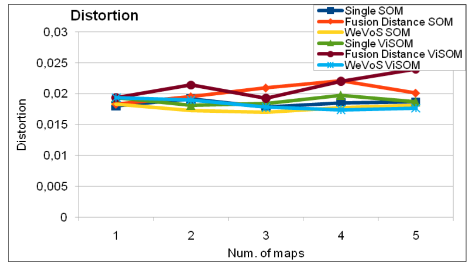
(e) Fusion by Distance applied to the ViSOM

(f) WeVoS-ViSOM

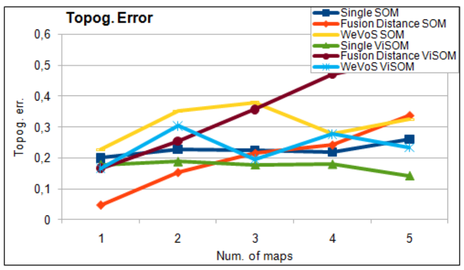
Figure 4: Maps obtained for the single models and two of the different summarization models described in this study, for both the SOM and ViSOM map training algorithms.



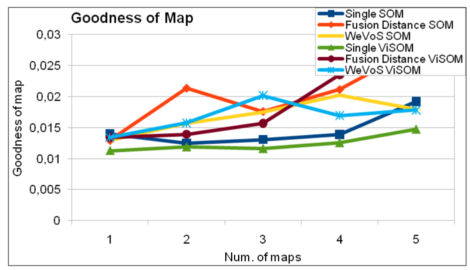
(a) Mean Quantization Error



(b) Distortion



(c) Topographic Error



(d) Goodness of Map

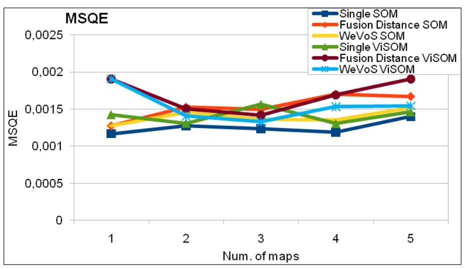
Figure 5: Evolution of Iris data set results when increasing the number of maps composing the ensemble.

6.3.1. Experiment 1

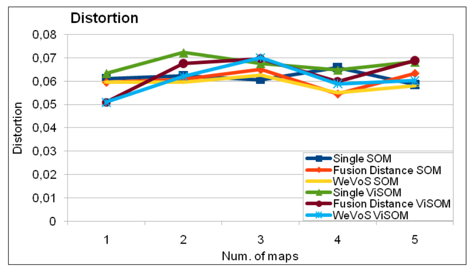
The first experiment consists in using the complete data set to train the ensembles; increasing the number of maps used to construct each of them, assessing the effect of the modification in the number of components of the ensemble.

Once the ensembles were trained, the fusion of the ensemble was computed by using the two variations explained in Section 4 and the novel algorithm presented in Section 5. In all cases, the weight of the vote for each unit in this latter model was calculated according to the goodness-of-adaptation measure. All the measures were calculated using the test part of the dataset, both for the average measure for the ensemble and for all the variations of the fusions of the ensemble. In the figures shown, ordinate axes represent the value of the error measure, while abscissa axes represent the number of composing models are used by the fusion algorithm.

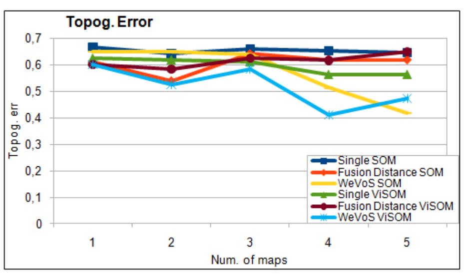
In the case of the iris data set (Fig. 5) the different ensemble models do not seem to introduce very interesting improvement for the quality measures calculated. The exception to this is the Distortion (Fig. 5b), in which both the WeVoS-SOM and WeVoS-ViSOM obtain better results than the single models and the other fusion algorithms; although it is not a very significant improvement. In the Mean Quantization Error (Fig. 5a) and the Goodness of maps (Fig. 5d) the best performing algorithms are the single models, being clearly the ViSOM the best of the two.



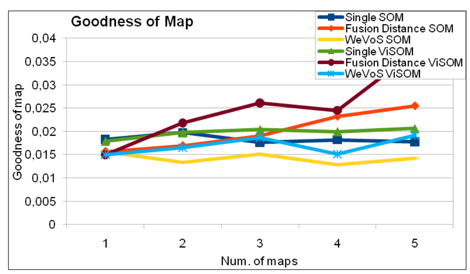
(a) Mean Quantization Error



(b) Distortion

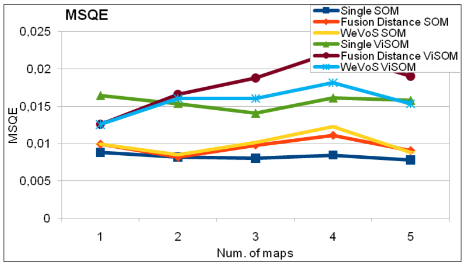


(c) Topographic Error

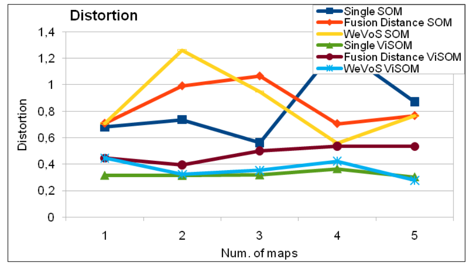


(d) Goodness of Map

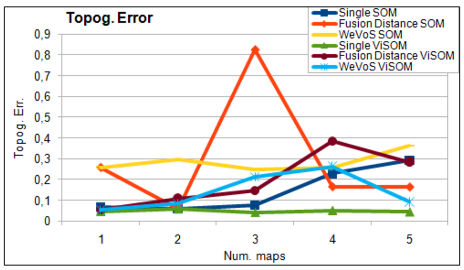
Figure 6: Evolution of Wine data set results when increasing the number of maps composing the ensemble.



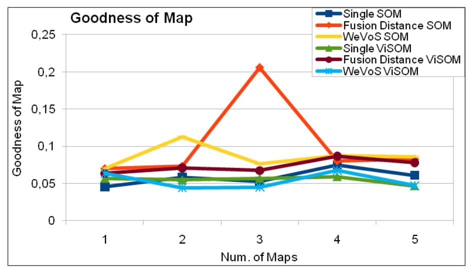
(a) Mean Quantization Error



(b) Distortion



(c) Topographic Error



(d) Goodness of Map

Figure 7: Evolution of Echo-Cardiogram data set results when increasing the number of maps composing the ensemble.

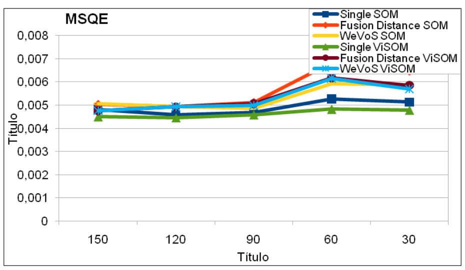
Regarding the Wine data set (Fig. 6), this situation changes. Although for the quantization error (Fig. 6a) the best models seem to be the single ones, for the other three measures (Figs. 6c, 6b and 6d) the ensemble algorithms are the best performing ones, especially in the topographic error (Fig. 6c) and the goodness of map (Fig. 6d). In this last measure, although the WeVoS-ViSOM outperforms the single ViSOM, the best algorithm is clearly the single SOM.

For the Echo-Cardiogram data set (Fig. 7) the results seem to be similar those of the Wine. The quantization error (Fig. 7a) is higher in the ensemble fusion algorithms. For Distortion (Fig. 7b), ViSOM and WeVoS-ViSOM are very close to each other although the single ViSOM seems to obtain slightly better results. For the other two measures (Figs. 7c and 7d) the WeVoS-ViSOM obtains the best results, also having low variations between results, which indicates the stability of the algorithm.

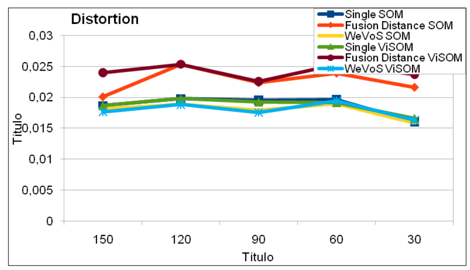
6.3.2. Experiment 2

The second experiment consists in using a moderated number of ensemble components but modifying the number of data samples used for the training of the models. This emulates the addition of noise or instability to the data sets, as when using less amount of data but maintaining its dimensionality the training process becomes more difficult.

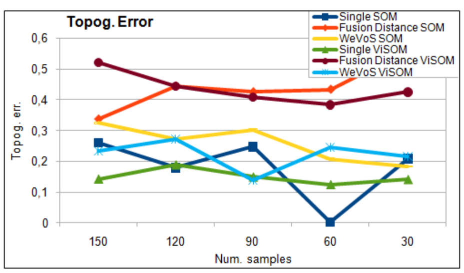
Results for this experiment confirm the results obtained in the previous one.



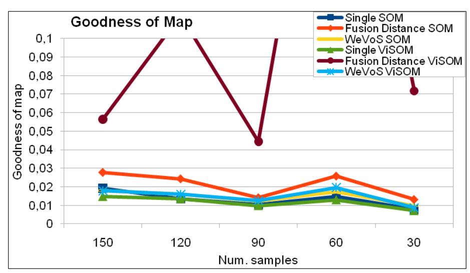
(a) Mean Quantization Error



(b) Distortion

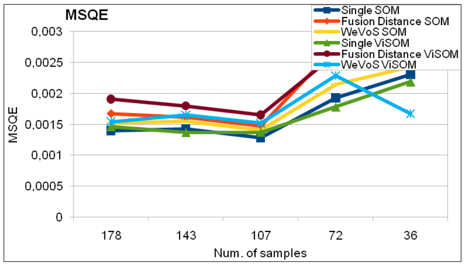


(c) Topographic Error

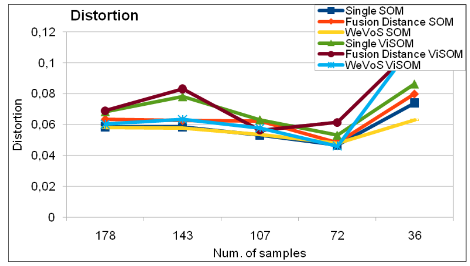


(d) Goodness of Map

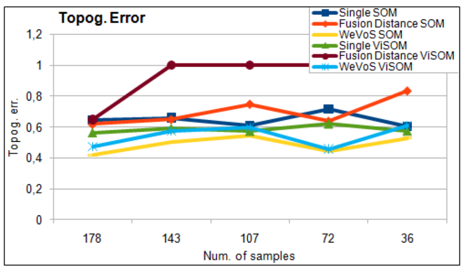
Figure 8: Evolution of the Iris data set results when decreasing the size of the samples set used of training.



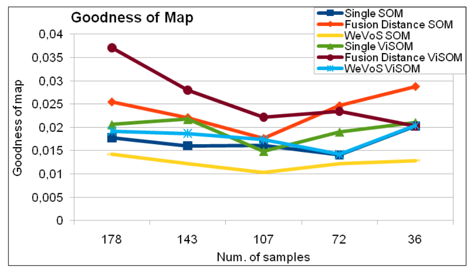
(a) Mean Quantization Error



(b) Distortion



(c) Topographic Error



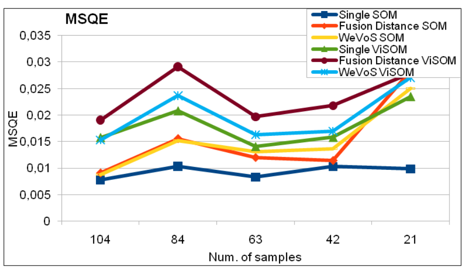
(d) Goodness of Map

Figure 9: Evolution of the Wine data set results when decreasing the size of the samples set used of training.

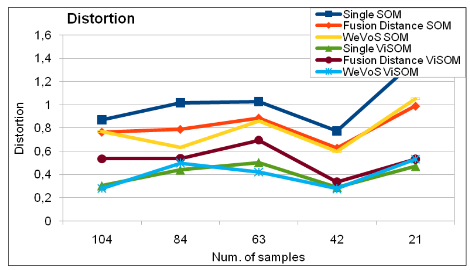
In the case of the Iris data set (Fig. 8) the WeVoS algorithm is not able to improve Quantization Error results (Fig. 8a), but it outperforms single models in the Distortion (Fig. 8b) and Topographic Error (Fig. 8c) measures, especially the WeVoS-ViSOM is the one obtaining the lowest error. For the Goodness of Map (Fig. 8d) all models -except Fusion by Distance algorithms- behave in a very similar way, being difficult to outline one model over the rest.

In the experiment performed with the Wine data set the results (Fig. 9) are also better for the ensemble algorithms using the WeVoS. Except in this case, although both the WeVoS-SOM and WeVoS-ViSOM algorithms outperform their single homologous algorithm; the one obtaining lower error turns out to be the WeVoS-SOM. This is especially true for the Goodness of Map (Fig. 9d) but to a minor extent is similar to the Topographic Error (Fig. 9c). For the Quantization Error (Fig. 9a), the ensemble algorithm still yields not as good results as single models.

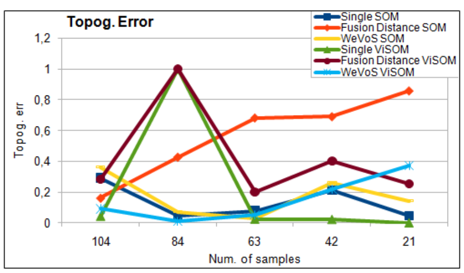
Finally, the last experiment, using the Echo-Cardiogram data set (Fig. 10), is the one with more distinguishing results. In this case is clear that ensemble models obtain higher Quantization Errors (Fig. 10a) than the single models. This is expected, as it is consistently true in all experiments. For the Distortion measure (Fig. 10b), clearly the best model is the WeVoS-ViSOM, although very close to the regular ViSOM. On its hand the WeVoS-SOM clearly outperforms the single SOM. For the Topographic Error (Fig. 10c), is the WeVoS-ViSOM



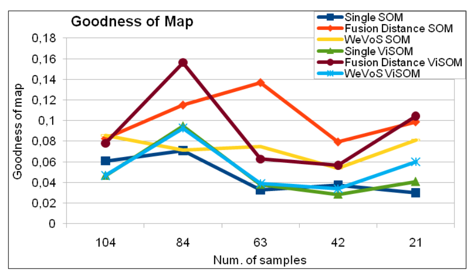
(a) Mean Quantization Error



(b) Distortion



(c) Topographic Error



(d) Goodness of Map

Figure 10: Evolution of the Echo-Cardiogram data set results when decreasing the size of the samples set used of training.

the one which clearly obtains the best results. In the Goodness of Map measure (Fig. 10d), the best performing models seem to be the single versions, especially the single SOM, which shows in this case a lower error than the WeVoS-SOM. The WeVoS-ViSOM and single ViSOM exhibit not such a good behaviour, but similar one to the SOM; although ViSOM performs slightly better when the size of the data set reduces to less than half of the original size.

6.4. Discussion

The results included show some clear conclusions about the Fusion of Topology Preserving Mapping algorithms. One is that, although it sometimes the Fusion by Euclidean Distance can show a better classification performance than the single SOM (as Georgakis et al. [10] demonstrates) this could rather be due to the effect of the re-labeling of neurons than the improvement of its topological characteristics. Also, as results show, the final structure obtained by this fusion algorithm is clearly not suitable for best representation of the data set structure, due to the twists appearing in the map grid. On the contrary, the WeVoS scheme shows a much regular grid, which as can be seen in the example presented with the iris (Fig. 2 and 3) can serve to better adjust the grid and distribute its units on the data input space.

Regarding the analytical results obtained, all favour this idea that although the Fusion by Distance can obtain better classification results, the visualization characteristics of the resulting maps are generally poor.

Among the models compared, the WeVoS-ViSOM is the one showing the best adaptation to the Iris data set, spreading the grid in a wider way over the data manifold. This translates into a better final visualization of the data set structure -as can be seen in Fig. 4- due to the enhanced visualization capabilities of the ViSOM and the added improvement of the WeVoS fusion algorithm.

For the WeVoS-ViSOM results prove some characteristics of the models that are interesting to note. First, as all experiments point out, the reduction of the quantization error is not the main interest of this algorithm. As is easy to see on Figs. 2 and 3 -but also on each analytical result - the algorithm tries to better spread its units along the input data space, rather than concentrate them to where more amount of data is located to get a more informative representation of the data space. This come to the cost of obtaining higher quantization error than other models. Concerning the other quality measures, the most interesting characteristic is that the usefulness of the WeVoS-ViSOM model for data visualization depends on the data set. As can be in Fig. 5, results for the Iris data set are not as good as for single models -with exception of the Distortion measure-. On Fig. 6 can be seen that best performing models are WeVoS-SOM (Distortion and Goodness of Map measures) and WeVoS-ViSOM (Topographic Error measure). And, finally, on Fig. 7 the WeVoS-ViSOM obtain some of best results (especially Topographic Error and Goodness of Map). These results are similar in the case of the second experiment (Figs. 8, 9 and 10).This points to the idea that for the ensemble to be really useful, the data set must have enough complexity from the point of view of an automated learning algorithm. For example: Iris data set has 150 samples, but only 4 dimensions, while Wine data set has 178 samples and 13 dimensions and the Echo-Cardiogram data set has 105 samples and 9 dimensions. In this case, as for classical classification ensembles; when a single algorithm performs in a correct way for a given data set, the ensemble fusion algorithms are not able to outperform it; while when the data set is complex for the single model, the use on an ensemble meta-algorithm is able to further improve the capabilities of the single one.

7. Conclusions and Future Work

In this research a novel topology preserving model called WeVoS-ViSOM is presented, analysed and compared with other models. This model aims to generate the most accurate visual representation of a multi-variable data set in the form of a 2D map that summarizes visually the principal features of the data set outlined by the different trained maps composing the ensemble. Its main objective is to obtain the most comprehensive visualization as possible, sacrificing the less as possible the topological presentation of the data, one of the main qualities of the Self-Organizing Maps. The main characteristic of the model is the smooth adaptation of to the input space of the data set, correcting small defects that can arise on a single training; and therefore further improving the visualization capabilities of the ViSOM algorithm. The present work has included detailed descriptions of previously devised summarization algorithms and compared them with the new model. The performance of the summaries obtained by the WeVoS meta-algorithm has been analysed by means of a range of quality measures; and the usefulness of the WeVoS-ViSOM has been proved empirically, showing that provides clearer and smoother representation of the inner structure of the data set under study. Although it does not outperform single models regarding its classification accuracy or quantization error, it succeeds in reducing the distortion error of single models, thus obtaining a more truthful and organized representation of the data set. In the cases analysed, the WeVoS-ViSOM has obtained lower errors than the WeVoS-SOM, proving a very useful tool for data visualization. Future work will be focused on the application of the WeVoS to other topology preserving models and to other cases of study. Also some improvements on the way the ensemble is calculated, taken from ensemble meta-algorithms most spread practices, will be tested in a wider array of real-life problems.

Acknowledgments

This research has been partially supported through projects CIT-020000-2008-2 and CIT-020000-2009-12 of the Spanish Ministry of Education and Innovation and BU006A08 of the Junta of Castilla and León. The authors would also like to thank the manufacturer of components for vehicle interiors, Grupo Antolín Ingeniería, S.A. in the framework of the project MAGNO2008-1028 CENIT. Project funded by the Spanish Ministry of Science and Innovation.

[1] Asuncion, A., Newman, D. J., 2007. UCI machine learning repository.

[2] Aurenhammer, F., Klein, R., 2000. Voronoi diagrams. In: Handbook of Computational Geometry. No. 5. North-Holland, Amsterdam, Netherlands, pp. 201–290.

[3] Bakker, B., Heskes, T., 2003. Clustering ensembles of neural network models. Neural Networks 16(2), 261–269.

[4] Baruque, B., Corchado, E., 2010. A weighted voting summarization of SOM ensembles. Data Mining and Knowledge Discovery 21, 398–426, 10.1007/s10618-009-0160-3.
URL http://dx.doi.org/10.1007/s10618-009-0160-3

[5] Baruque, B., Corchado, E., Yin, H., 2007. ViSOM ensembles for visualization and classification. In: 9th International Work-Conference on Artificial Neural Networks (IWANN'07). Vol. San Sebastian, Spain of Lecture Notes in Computer Science. Springer Berlin / Heidelberg, pp. 235–243.

[6] Baruque, B., Corchado, E., Yin, H., Rovira, J., González, J., 2007. Automated ham quality classification using ensemble unsupervised mapping models. In: Apolloni, B., Howlett, R. J., Jain, L. (Eds.), Lecture Notes in Computer Science. Vol. Vietri Sul Mare, Italy. Springer Berlin / Heidelberg, pp. 435–443.

[7] Breiman, L., 1996. Bagging predictors. Machine Learning 24 (2), 123–140.

[8] Corchado, E., Baruque, B., Yin, H., 2007. Boosting unsupervised competitive learning ensembles. In: 17th International Conference in Artificial Neural Networks (ICANN 2007). Vol. Porto, Portugal of Lecture Notes in Computer Science. Springer Berlin / Heidelberg, pp. 339–348.

[9] Dietterich, T. G., 2000. Ensemble methods in machine learning. In: MCS '00: Proceedings of the First International Workshop on Multiple Classifier Systems. Springer-Verlag, London, UK, pp. 1–15.

[10] Georgakis, A., Li, H., Gordan, M., 2005. An ensemble of SOM networks for document organization and retrieval. In: International Conference on Adaptive Knowledge Representation and Reasoning (AKRR'05). pp. 6–141.

[11] Gou, H.-T., Xu, J.-S., Wang, L., aug. 2006. A new clustering method based on references, density and neural network. pp. 1060 –1065.

[12] Henriques, A. P. M., Neto, A. D. D., Amaral, R. F., 2010. Classification of multispectral images in coral environments using a hybrid of classifier ensembles. Neurocomputing 73 (7-9), 1256–1264.

[13] Heskes, T., 1997. Balancing between bagging and bumping. In: Mozer, M. C., Jordan, M. I., Petsche, T. (Eds.), Advances in Neural Information Processing Systems. Denver, Colorado, USA, pp. 466–472.

[14] Hotelling, H., 1933. Analysis of a complex of statistical variables into principal components. Journal of Education Psychology 24, 417–444.

[15] Johansson, U., Lofstrom, T., Niklasson, L., 2006. Obtaining accurate neural network ensembles. In: Mohammadian, M. (Ed.), International Conference on Computational Intelligence for Modelling, Control & Automation Jointly with International Conference on Intelligent Agents, Web Technologies & Internet Commerce, Vol 2, Proceedings. pp. 103–108.

[16] Kaski, S., Lagus, K., 1996. Comparing self-organizing maps. In: Malsburg, C. v. d., Seelen, W. v., Vorbruggen, J. C., Sendhoff, B. (Eds.), International Conference on Artificial Neural Networks (ICAAN'96). Vol. Bochum, Germany of Lecture Notes in Computer Science. Springer, Berlin, pp. 809–814.

[17] Kiviluoto, K., 1996. Topology preservation in self-organizing maps. In: IEEE International Conference on Neural Networks (ICNN'96). Vol. 1. pp. 294–299.

[18] Kohonen, T., 1974. Adaptive associative memory principle. Ieee Transactions on Computers C 23 (4), 444–445.

[19] Kohonen, T., 1995. Self-Organizing Maps. Vol. 30. Springer, Berlin, Germany.

[20] Kohonen, T., 2008. Data management by self-organizing maps. In: WCCI. pp. 309–332.

[21] Kohonen, T., Lehtio, P., Rovamo, J., Hyvarinen, J., Bry, K., Vainio, L., 1977. A principle of neural associative memory. Neuroscience 2 (6), 1065–1076.

[22] Kuncheva, L. I., 2004. Combining Pattern Classifiers: Methods and Algorithms. Wiley-Interscience.

[23] Petrakieva, L., Fyfe, C., 2003. Bagging and bumping self organising maps. Computing and Information Systems Journal, –.

[24] Polani, D., 2003. Measures for the organization of self-organizing maps. In: Seiffert, U., Jain, L. C. (Eds.), Self-organizing Neural Networks: Recent Advances and Applications (Studies in Fuzziness and Soft Computing). Vol. 16. Physica-Verlag, Heidelberg, pp. 13–44.

[25] Pozlbauer, G., 2004. Survey and comparison of quality measures for self-organizing maps. In: Paralic, J., Polzlbauer, G., Andreas, R. (Eds.), Fifth Workshop on Data Analysis (WDA'04). Elfa Academic Press, pp. 67—82.

[26] Ron, M., Gunnar, R., 2003. An Introduction to Boosting and Leveraging.

[27] Ruta, D., Gabrys, B., 2005. Classifier selection for majority voting. Information Fusion 6 (1), 63–81.

[28] Saavedra, C., Salas, R., Moreno, S., Allende, H., 2007. Fusion of self organizing maps. In: 9th International Work-Conference on Artificial Neural Networks (IWANN 2007). pp. 227–234.

[29] Schwenk, H., Bengio, Y., 2000. Boosting neural networks. Neural Computation 12 (8), 1869–1887.

[30] Vesanto, J., 1997. Data mining techniques based on the self-organizing map. Ph.D. thesis, Helsinki University of Technology, Espoo, Finland.

[31] Vesanto, J., Sulkava, M., Hollmen, J., 2003. On the decomposition of the self-organizing map distortion measure. In: Proceedings of the Workshop on Self-Organizing Maps (WSOM'03). pp. 11–16.

[32] Wang, F., Yang, C., Lin, Z., Li, Y., Yuan, Y., 2010. Hybrid sampling on mutual information entropy-based clustering ensembles for optimizations. Neurocomputing 73 (7-9), 1457–1464.

[33] Yin, H., 2002. Data visualisation and manifold mapping using the ViSOM. Neural Networks 15 (8-9), 1005–1016.

[34] Yin, H., 2002. ViSOM - a novel method for multivariate data projection and structure visualization. Neural Networks, IEEE Transactions on 13 (1), 237–243.