



are far less widespread. In the case of this study, it is performed at the level of the dataset and at the level of the training of the algorithm. The first consists of adding data that has not already been labeled to provide additional information into the training, by repeating the training without and with the additional data. The second consists of re-sampling the dataset; so it simulates several bootstrap replications, and it trains a model on each of them and finally fuses all into a final mapping.

This may be due to the fact that it is a more complex task to apply classical ensemble algorithms without having quantitative assessment on how well an algorithm is performing at each step with unsupervised learning. In the case of the topology-preserving networks it is even more complicated, as results of the analysis are especially aimed at visual inspection of data, which is inherently subjective.

This study aims to expand the results obtained in the previous studies with the topology-preserving algorithms. Although there are many strengths in the way these algorithms function, like the lack of need for an external supervisor or specific prior knowledge of the dataset or their tolerance to faults, one of the most criticized weaknesses is the instability of their training. Running the algorithm over the same data, even with the same parameters, will not necessarily yield the same results. It is important to have consistent results to perform a confident analysis of the data in practice. In this sense, it is essential to obtain or assure in some way that the representation obtained is not the result of a random combination of factors in one determined run of the algorithm.

One of the most common techniques to improve the classification accuracy and stability of the algorithm is the replication of the training. By performing several replications of the training on the same data or their bootstrapped samples, the common or more important features can be identified, as they will appear in the majority of the runs. Then, those features can be fused in a final model that will perform the classification, often outperforming the single models.

This fusion of the replicated runs can be performed at several levels: at data level by including algorithms trained in different datasets; at output level, by averaging the classification results of differently trained classifiers; at algorithm level, by fusing several

classifiers; etc. In the case of this study, it is performed at the level of the dataset and at the level of the training of the algorithm. The first consists of adding data that has not already been labeled to provide additional information into the training, by repeating the training without and with the additional data. The second consists of re-sampling the dataset; so it simulates several bootstrap replications, and it trains a model on each of them and finally fuses all into a final mapping.

The remainder of the paper is organized as follows: Section 2 briefly summarizes the most usual types of learning neural networks, with emphasis on the interest of this study: unsupervised learning. The family of models used in the present study is then described in Section 3, whilst Section 4 presents a more recent approach to unsupervised learning used in this study. Section 5 introduces the other novelty included: i.e. the fusion algorithms for ensembles of topology preserving maps, followed by Section 6 on the proposed novel algorithm in this study (S<sup>2</sup>-Ensemble Fusion Algorithm), combining both previously explained approaches. Section 7 describes the experiments performed, the results and conclusions obtained. Finally Section 8 concludes the study and outlines future lines of research.

## 2. Unsupervised Learning

There are two main classes of learning algorithms for automated weight setting in neural networks: supervised and unsupervised learning. Humans seem able to learn certain tasks without explicit supervision. One aim of unsupervised learning algorithm is to mimic this aspect and thus this type of learning is arguably more biologically plausible than the error correction or gradient descent methods. For example such algorithms have local processing at each synapse with no global information passing being necessary. So an unsupervised network must self-organize with respect to its internal parameters, without external prompting, and to do so, it must react to some aspect of the input data. Typically there will be either redundancy or clusters in the data; i.e. some structure in the data to which it can respond to. There are two major components of

unsupervised learning: Hebbian learning and Competitive learning.

This study is focused on Competitive Learning where the output neurons of a neural network compete among themselves for being the active (firing) one. This mirrors the biological neurons in that there are finite resources for learning and so one neuron's gain means another's loss. This is different from Hebbian learning in which several output neurons may be active simultaneously; in the case of competitive learning only one output neuron is active at a time. This characteristic makes competitive learning a highly suitable tool to find those statistically salient features that may be used to classify a set of input patterns. Some examples of competitive learning used in unsupervised learning are the Self-Organizing Maps and its variants such as ViSOM.

### 3. Topology Preserving Models

Topology preserving mapping comprises a family of techniques with a common target: to produce a low dimensional representation of the training samples while preserving the topological properties of the input space. The main application of these algorithms is a straightforward data visualization or representation that can serve for dataset inspection by users or as pre-processing tasks for other models, such as perception or planning algorithms.

The two models used in this study are two of the most wide-spread members of this family, but still they are devised with very different representation objectives and can obtain quite different results. Both of them have been used separately in previous related studies.

They are used in this case to compare directly how the proposed enhancing meta-algorithms behave under the frame of different competitive learning algorithms.

### 3.1. Self-Organizing Map

The best known technique among topology preserving algorithms is the Self-Organizing Map (SOM) algorithm. It was originally devised to obtain a representation of a multi-dimensional dataset, on a low dimensional grid or lattice.

It is based on competitive learning as well as Hebbian learning and is an adaptive process in which the neurons in a neural network gradually become sensitive to different input categories from sets of samples in a specific domain of the input space.

As a result of the learning process, i.e. the presentation of all input vectors and the adaptation of the weight vectors, the SOM generates a mapping from the input space onto the lattice,  $U$ , in such a way that the topological relationships of the input space are as faithfully as possible preserved on  $U$ .

If not only the winning neuron but also its neighbours on the lattice are allowed to learn, that is to adapt its characteristics to those presented as the input; then, neighboring neurons gradually specialize to represent similar inputs, and the representations become ordered on the map lattice. This is the main feature of the SOM algorithm. The update of neighborhood neurons in SOM is performed as follows:

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

Where  $t$  denotes a given iteration of the algorithm,  $w_k$  is the weight of a neuron,  $k$ , within the neighborhood of the "best matching unit" (BMU),  $v$ ;  $\alpha$  is the learning rate of the algorithm;  $\eta(v,k,t)$  is the neighborhood function (usually a Gaussian function); and  $x$  is the input.

### 3.2. Growing Neural Gas

The Growing Neural Gas (GNG) is a representation and clustering algorithm proposed by Fritzke, based on the neural gas (NG) algorithm previously proposed by Martinetz *et al.* for finding optimal data representations. Both NG and GNG are modifications of the SOM algorithm. The main characteristic of the NG algorithm is that instead of expanding through the data input space as a fixed grid of units, it allows the

neighboring relationships of its units to change, expanding more like a gas over the data space.

The GNG is different from the previous algorithms in that it is an incremental algorithm, so there is no need to determine a priori the number of nodes. Network shape and size are determined during the training, while the SOM and NG are often trained on a fixed network size throughout.

The GNG is a combination of Fritzke's Growing Cell Structures (GCS) and Martinetz's competitive Hebbian learning (CHL). In each step of the algorithm, the error of the units in representing the data is calculated as the Euclidean distance of units and inputs:

$$\Delta error(s_1) = \|w_{s_1} - x\|^2 \quad (2)$$

where  $s_1$  is the closest unit to the input entry  $x$  and  $w_{s_1}$  is the weights vector corresponding to that unit.

The network topology of the GNG is generated incrementally by the CHL algorithm, which successively inserts topological connections or edges in the points where the error is higher. The main principle of the CHL is: for each input  $x$  connect the two closest centers (measured by Euclidean distance) by an edge. Then, the weights of the units are updated, for the BMU ( $w_{s_1}$ ) and for their neighbors ( $w_n$ ) as:

$$\begin{aligned} w_{s_1} &= e_b(x - w_{s_1}) \\ w_n &= e_n(x - w_n) \end{aligned} \quad (3)$$

### 3.3. Quality Assessment Measures

In the case of supervised learning, having established a collection of desired outputs for the inputs means that the deviation of the learning state from the desired output can be calculated in a rather straightforward way. When dealing with unsupervised learning, the process only depends on the inputs and the dynamics of the learning rule; making it much more difficult to determine the accuracy of the training in relation to the input dataset.

There is no single, general canonical measure to determine the quality of the training of unsupervised learning algorithms, including topology preserving models. This family of algorithms is able to perform several different tasks; such as data visualization,

clustering or pattern matching. That is why so many different measures are used in the literature, depending on the function that is intended for the algorithm under study. There is no "best" measure, but rather several complementary ones.

In this study three common quality measures applicable to both the SOM and GNG have been chosen, in order to quantify how the meta-algorithms proposed can improve the base models in more than one of their performing areas.

#### 3.3.1. Classification Error

Classification error is calculated as a percentage of the number of samples that are wrongly classified by the algorithm with respect to the total number of samples given to the classifier. Although the topology preserving maps algorithms are not designed for classification purposes; this measure can serve, to some extent, for assessing the quality of the training.

This is achieved by first training the model on the training dataset and then, presenting again the dataset to the model. Then a record is kept for each neuron of how many times it reacted to a determined class of data. After this process it can be considered that the neuron is specialized in recognizing samples of that same class.

It is reasonable to think that a map that consistently recognizes samples in a correct way is better adapted to the dataset in which it has been trained than another map that does not obtain consistent results and, therefore, will represent better the structure of the data. This correct identification and separation of samples will ultimately produce better representation or visualization results.

This measure has been employed in several previous studies involving some of the techniques in this research.

#### 3.3.2. Quantization Error

Quantization Error is related to all forms of vector quantization and clustering algorithms. Thus, it disregards map topology and alignment. Quantization error is computed by determining the average distance of the data entries to the cluster centroids represented the learnt neurons. In case of the SOM, the cluster centroids are the characteristics or weight vectors of the neurons. Its mathematical expression is:

$$E_Q = \frac{1}{|D|} \sum_{x_i \in D} \|x_i - w_i\|^2 \quad (4)$$

where  $|D|$  is the number of entries in the dataset used  $D$ , and  $w_i$  the BMU for each input  $x_i$ .

The measure tells how well the model approximates the dataset under study. That is, it accounts for the overall distance between data samples and the neurons representing them.

### 3.3.3. Goodness of Map

This measure, described by Kaski and Lagus<sup>¡Error! No se encuentra el origen de la referencia.</sup> combines two error measures: the squared quantization error and the topographic error. It takes into account both the distance between the input and the BMU and the distance between the first BMU and the second BMU on the shortest path along the map grid. It measures both the continuity of the mapping from the dataset to the map grid, and the accuracy of the map in representing the set. The mathematical expression of this measure for each data entry is shown below:

$$d(x_i) = \|x_i - v_i\| + \min \sum_{k=0}^{|k_{v_i}|-1} \|w_{I_i(k)} - w_{I_i(k+1)}\| \quad (5)$$

where,  $v_i$  and  $v'_i$  represent the weights of the first BMU and the second BMU respectively, corresponding to data entry  $x_i$ .  $I_i(k)$  and  $I_i(k+1)$  represent indexes of the  $k^{th}$  and the  $k^{th+1}$  neurons along the minimum path from  $v_i$  to  $v'_i$ , both neurons being direct neighbors on the map grid. By definition  $w_{I_i(0)} = w_{v_i}$ ; which is to say, the first neuron in the path is the first BMU for data entry  $x_i$  and  $w'_{I_i(k_{v_i})} = w_{v'_i}$ , i.e., the last neuron in the path, corresponding to the second BMU for  $x_i$ . The final goodness of the map is defined as the average of the values of Eq. 5 for all data entries in the dataset.

This measure is the most comprehensive of the three, as it accounts for both the quantization of the map and the organization of the grid, penalizing close units in the input space that are not neighbors in the output space.

## 4. Semi-Supervised Learning

As several recent studies point out,<sup>¡Error! No se encuentra el origen de la referencia.</sup><sup>¡Error! No se encuentra el origen de la referencia.</sup> the use of unlabeled data has proven to be a useful way of

improving the classification built on previously labeled datasets. This is especially useful when adding label information to new datasets is expensive or difficult in practical situations. Thus, characteristics of new unclassified samples can be incorporated into the classifier in a meaningful way.<sup>¡Error! No se encuentra el origen de la referencia.</sup><sup>¡Error! No se encuentra el origen de la referencia.</sup>

One of the simplest methods of incorporating unlabeled data is the so-called semi-supervised learning. It consists of presenting the unlabeled data to an already trained classifier. The classifier first tries to determine the corresponding labels for the unlabeled samples and then incorporates this new knowledge to its re-training. It is a seemingly straightforward method, as it relies only on the classifier capabilities.

Other widespread methods where unlabeled-data information is incorporated to the learning process is the active learning. This modification of the classical supervised learning also uses a pool or stream of unlabeled samples to improve the learning results. It works by determining in some way which of the unlabeled samples are more informative for the problem at hand, then consulting an “oracle” about the class of that samples and finally incorporating this new information to the training.<sup>¡Error! No se encuentra el origen de la referencia.</sup><sup>¡Error! No se encuentra el origen de la referencia.</sup> That way, only “interesting” information needs to be labeled; reducing the number of overlapping or non-informative samples that have to be labeled. This is a very useful technique in cases where the labeling is a complex task for the users such as: speech recognition, information extraction, image classification, etc.<sup>¡Error! No se encuentra el origen de la referencia.</sup><sup>¡Error! No se encuentra el origen de la referencia.</sup>

Obviously, the most difficult task in this case is determining which the most interesting samples to label are. There are several techniques and studies on that subject.<sup>¡Error! No se encuentra el origen de la referencia.</sup> Active learning has been left out of this study for the sake of simplicity.

This study will therefore make use of the semi-supervised algorithm proposed in a previous study.<sup>¡Error! No se encuentra el origen de la referencia.</sup> This is a two-stage method where labeled data are firstly used to train a classifier and then unlabeled data are labeled according to the classifier trained with the original labeled data. The second stage involves classifying unlabeled data and re-training the classifier from the classified unlabeled data as well as the original labeled data. The two stages are

iterated until the training process converges, in other words, the training errors stabilize.

This procedure can be applied to many different classifiers. Some recent studies combine this kind of learning with topology preserving maps with interesting results.<sup>[Error! No se encuentra el origen de la referencia.], [Error! No se encuentra el origen de la referencia.].</sup> In previous studies,<sup>[Error! No se encuentra el origen de la referencia.], [Error! No se encuentra el origen de la referencia.].</sup> the algorithm is proved to increase the quality of the classification results of the GNG. In this study it is applied to the SOM and GNG and their ensemble variations to test to what extent the use of this algorithm in combination with ensemble techniques can further improve the performance.

## 5. Ensembles and Fusion Algorithms

The use of an ensemble of similarly trained models or algorithms was conceived to improve the performance of classification algorithms.<sup>[Error! No se encuentra el origen de la referencia.], [Error! No se encuentra el origen de la referencia.].</sup> It has been observed that, although one of the classifiers in an ensemble may yield the best performance, the sets of patterns that are misclassified by the different classifiers would not necessarily overlap.<sup>[Error! No se encuentra el origen de la referencia.], [Error! No se encuentra el origen de la referencia.].</sup> This suggests that 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 idea is not to rely on a single decision making scheme, but to use all the designs, or their subsets, by combining their individual opinions to derive a consensus decision.<sup>[Error! No se encuentra el origen de la referencia.], [Error! No se encuentra el origen de la referencia.].</sup>

In this study, the conceptual perspective that we follow to improve the performance of the model is that of a single “summary” or “synthesis” of the inputs stored within the whole ensemble, that is referred to as ‘fusion’ throughout this research. The main objective is to obtain a unique map that can be seen to represent as clearly and as reliably as possible the different features contained in the different maps in the ensemble.

### 5.1. Voronoi Polygons Similarity Fusion

The ultimate goal of constructing an ensemble is to improve the performance obtained by a single working unit.<sup>[Error! No se encuentra el origen de la referencia.], [Error! No se encuentra el origen de la referencia.].</sup> In the present study, the central idea is to verify the improvements that an ensemble technique can provide in the multi-dimensional data classification over a semi-supervised

learning process performed with the competitive learning.

As the initial aim of the ensemble architecture was to improve supervised classification, few models attempt to deal with ensembles of unsupervised learning, and even fewer try to deal with the topology preserving maps.<sup>[Error! No se encuentra el origen de la referencia.], [Error! No se encuentra el origen de la referencia.].</sup> In this study, an algorithm for topology preserving networks summarization proposed by Saavedra et al.<sup>[Error! No se encuentra el origen de la referencia.], [Error! No se encuentra el origen de la referencia.].</sup> is used as a potential mean of improving the single model's performance. This method was selected as it is the one of the algorithms that deal best with the GNG due to its particular way of constructing its growing net as a graph of connected nodes.<sup>[Error! No se encuentra el origen de la referencia.], [Error! No se encuentra el origen de la referencia.].</sup> Other algorithms, though also applicable, are more dedicated to maintain a map grid.

This algorithm uses the Voronoi polygons related to the units of different networks as a means of comparing them and deciding on the structure for the final fused network. Each unit in a topology preserving map can be associated with a portion of the input data space called the Voronoi polygon.<sup>[Error! No se encuentra el origen de la referencia.], [Error! No se encuentra el origen de la referencia.].</sup> That portion of the multi-dimensional input space contains data for which that precise unit is the BMU of the whole network. Thus, a logical conclusion is to consider that the units related to similar Voronoi polygons are considered similar as they are situated relatively close in the input data space. A record may be kept of which data entries activated each unit as the BMU, to calculate the dissimilarity between the Voronoi polygons of two units. This can easily be done by associating a binary vector with the unit, the length of which is the size of the dataset and contains zeros in the positions where the unit was the BMU for that sample and zeros in all other positions. The dissimilarity (i.e. the distance) between units can therefore be calculated, as shown in Eq. 6:

$$ds(b_r, b_q) = \frac{\sum_{l=1}^n XOR(b_r(l), b_q(l))}{\sum_{j=1}^n OR(b_r(j), b_q(j))} \quad (6)$$

where,  $r$  and  $q$  are the units whose dissimilarity will be determined and  $b_r$  and  $b_q$  the binary vector relating each unit with the data sample that it recognizes. All the bits in the vectors are used in this comparison.

The main issue with this proximity criterion is that it depends on data recognition by the network. To avoid problems of “dead” units, all units with a reacting rate lower than a set threshold are removed before calculating the similarities between them.

Units that are sufficiently similar are grouped together to form a single unit on the final map. Eq. 7 is used to determine the units that will be part of the same group or cluster,

$$S_i : \begin{cases} ds(b_r, b_q) > \theta_f, \forall r, q \in W_{S_n} \\ ds(b_r, b_q) < \theta_f, \forall r, q \notin W_{S_n} \end{cases} \quad (7)$$

where  $ds(b_r, b_q)$  is the dissimilarity between units  $b_r$  and  $b_q$ , (see Eq. 6) and  $\theta_f$  is the fusion threshold.

Having determined the units to be fused, the final unit is obtained by calculating the centroid of the fused units (Eq. 8):

$$w_c = \frac{1}{|W_k|} \sum_{w_i \in W_k} w_i \quad (8)$$

Finally, the similarity criteria must be used again to keep a notion of neighboring relations between the units of the fused network. Units with dissimilarity below a certain threshold among their initial composing units are considered as neighbors in the fused network (Eq. 9):

$$\min_{(b_r \in W_{S_k}, b_q \in W_{S_l})} ds(b_r, b_q) < \theta_c \quad (9)$$

where  $\theta_c$  is the connection threshold of the algorithm. A detailed description of the algorithm can be found in the original publication.<sup>!Error! No se encuentra el origen de la referencia.</sup>

This whole procedure implies that the final neural architecture will approximate well the dataset, enhancing the vector quantization feature of the SOM. Its drawbacks are that the number of units in the final network is an unknown factor before its fusion (and will almost certainly differ from the size of the composing networks) and that the neighborhood relationships of the composing networks will be ignored in the final one, as the latter will create a new neighborhood for each unit based on its dissimilarity with the others.

## 6. The S<sup>2</sup>-Ensemble Fusion Algorithm

The novel model presented in this study enables the use of unlabeled data to improve the quality of the training by incorporating new but unlabeled data samples with

semi-supervised learning. At the same time, the learning procedure is repeated in parallel several times to add diversity to the resulting maps and finally the most interesting of the diverse features found are grouped together in a final map by means of a fusion algorithm. That way, the process benefits from the increase in the different samples that can be used for its training and from the several slightly different approaches that can be chosen from. A detailed description of the S<sup>2</sup>-Ensemble Fusion Algorithm is given in Algorithm 3.

### Algorithm 3 The S<sup>2</sup> Ensemble Fusion Algorithm

*Input:*  $L$  and  $U$ , let  $L' = \{\phi\}$  represent an initial empty set of newly labeled data.  $m$ : number of maps in the ensemble

*Output:* Classifier Algorithm ( $M_{fus}$ )

1. **Procedure** S<sup>2</sup> Ensemble Fusion  
( $L = (x, y)$   $U = x'$ :  $x, x' \in R^n$ ;  $y \in C$ )
2. **for**  $i = 1$  **to**  $m$  **do**
3. Generate  $L_i$  by re-sampling with replacement in  $L$ :  $L \rightarrow L_i$
4. Generate  $U_i$  by re-sampling with replacement in  $U$ :  $U \rightarrow U_i$
5. **end for**
6. **for**  $i = 1$  **to**  $m$  **do**
7. Train a map  $M_i$  with  $L_i$
8. Classify all instances of  $U_i$  using  $M_i \rightarrow L_i'$
9. Train a map  $M_i$  with  $L_i + L_i'$
10. Check if labels from  $U_i$  became stable
11. **end for**
12. **for all** maps in  $M_i$  **do**
13. Remove non-reacting units
14. Calculate dissimilarity between all units (Eq. 6)
15. Group units according to similarity (Eq. 7) and calculate centroids (Eq. 8)  $\rightarrow M_{fus}$
16. **end for**
17. Perform map Fusion by Similarity by connecting final units (Eq. 9)  $\rightarrow M_{fus}$
18. **end procedure**

As is intuitively seen from the pseudo-code, the meta-algorithm adds a high computational complexity to the regular training of a topology preserving mapping algorithm. The regular algorithm is repeated several times during the semi-supervised training and also this is repeated in parallel to obtain an ensemble.

It is therefore important to ensure that the extra effort of using this algorithm is required to obtain better results than those that would be obtained by the

classical simple algorithm. Validation and examples of this are extensively shown in the experiments and results (Section 7).

## 7. Experiments and Results

Experiments were performed using the well-known datasets from the UCI repository. They include: Iris, Cancer, Wine, Echo-cardiogram and Glass datasets. The first three are the most used in the literature. The former two were selected to highlight some characteristics of the presented models.

The iris dataset is a relatively small one (150 samples with 4 dimensions) with 3 quite separated classes. It is usually used as a simple test for different models. Comparatively, the Wisconsin Breast Cancer has also three much more intricate classes, but is much bigger (683 samples with 9 dimensions); so has more elements to extract characteristics. The Wine dataset has a similar size to the Iris (178 samples) but includes higher number of dimensions (13 dimensions) and thus is a more difficult dataset. According to this; the Iris dataset has much less complexity than the Wine dataset. The rest of the datasets used are in the middle of those examples.

The main process for the experiments is the same for both types. For calculating the numerical results, a 5-fold cross-validation is used. This means that the dataset is first divided into five folds of the same size. In each of the five repetitions of the experiment one fold is used as the test set. The other four folds are used as the training set. From these three quarters of the samples are considered labeled samples, and the rest are considered unlabeled. When training unsupervised models, only the labeled subset is used, while when using semi-supervised training, both labeled and unlabeled sets are used.

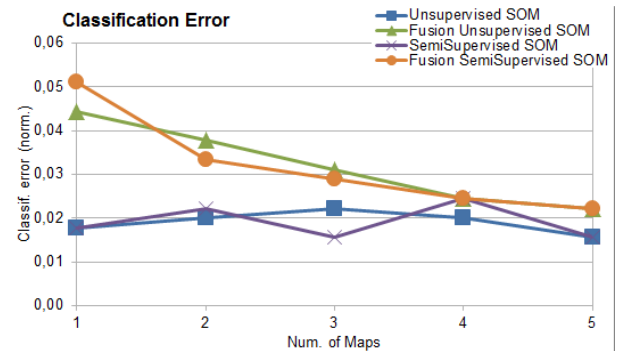
In all experiments the parameters used for the Voronoi Polygon Similarity Fusion are those appearing in Table 11. The parameters for the base classifiers in each experiment are detailed in Table 12.

In the following subsections, the experiments present error measures. This means that the closer a measure is to the abscissa axis, the better is the result considered. For the Quantization error and the Goodness of Map, values in the ordinate axis are absolute values, for the Classification Error, the values showed are the percentage of wrong classified samples normalized to 1.

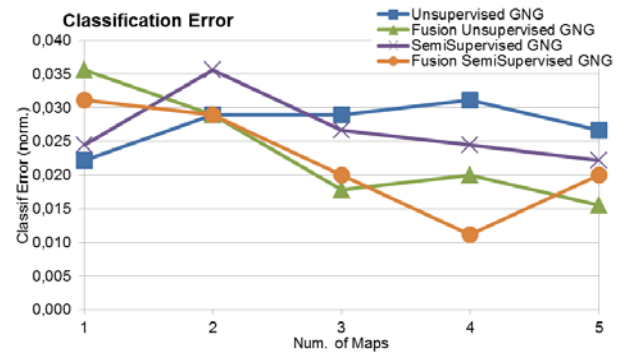
### 7.1. Experiment 1: Increasing the number of components of the ensemble

The first experiment consists of using the complete dataset to train ensembles of unsupervised or semi-supervised models; increasing the number of maps used to construct the ensemble, to assess the effect of the modification in the number of components 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 used by the fusion algorithm.

Results for the Iris dataset are presented in Figure 1 and Figure 2; while results for the Wine dataset are showed in Figure 3 and Figure 4. The numerical results of these datasets and the other three mentioned are included in Table 1, Table 4 and Table 5.



(a). Classification Error for the SOM-based models



(b). Classification Error for the GNG-based models

Figure 1. Comparative of the influence of the Semi-supervised Learning and Ensemble Fusion algorithms on the Classification Error of the studied models when analyzing the Iris dataset.



When analyzing the Iris dataset under the Classification Error perspective (shown in Fig. 1), the influence of the Semi-Supervised Learning is not something to have into account in combination with the SOM algorithm: Unsupervised and Semi-Supervised SOM obtain similar results whether combined with the single models or with the Fusion algorithm. In this case, also the use of Fusion obtains clear worse results.

When the experiment is run with the GNG as base classifiers, the situation is the opposite: Semi-Supervised Learning obtains better results than the classical Unsupervised GNG.

This seems due to the different nature of the training of both base algorithms. The GNG is able to develop a much freer structure, being able to modify the number of composing units and their neighboring. On its hand, the SOM is intended to keep a fixed grid. Therefore it is more difficult for the SOM map to modify the state it has achieved by training to adapt to the modified conditions on each of the iterations of the semi-supervised training.

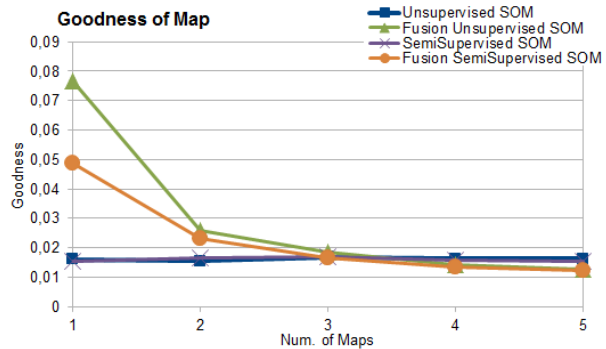
Learning and Ensemble Fusion algorithms on the Goodness of Map of the studied models when analyzing the Iris dataset.

The Fusion meta-algorithm obtains a clear improvement over single models (unsupervised or semi-supervised). This is due to the way the Fusion is performed. The Fusion meta-algorithm is better suited for the GNG, as it is a much freer structure, where the modification of neighborhood relationships and the number of units is part of its functioning.

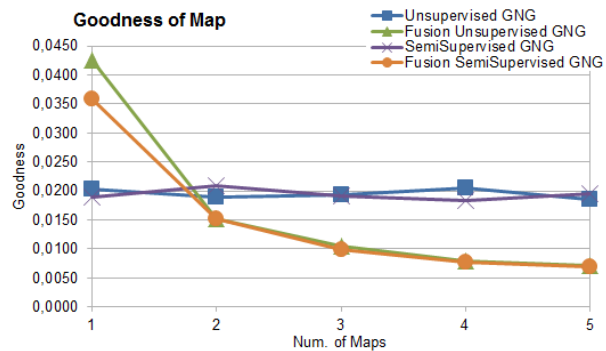
This is also true for the Cancer and Echo-cardiogram datasets (shown in Table 1) in which the single SOM algorithms obtain better results than the Fusion counterparts, but the Fusion GNG outperforms the single GNG – both Unsupervised or Semi-Supervised.

It is also interesting to note that the use or not of the semi-supervised learning does not affect the rest of the characteristics measured for topology-preserving maps – such as quantization or goodness. This was expected, as the algorithm only takes into account classification performances, so it does not affect other characteristics of the models.

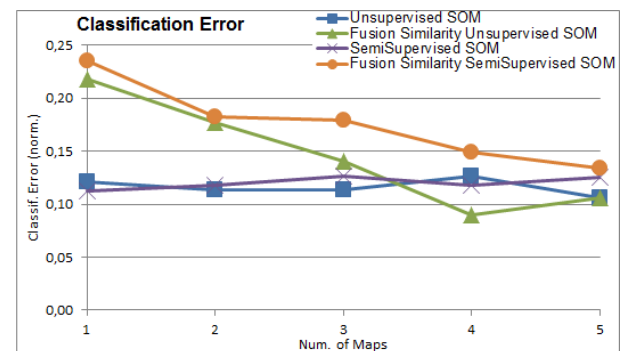
The Goodness of Map results obtained (Fig. 2) mirror very much what happens with Classification Error for the Unsupervised and Semi-Supervised variants under study: one does not clearly outperform the other. On the contrary, the Fusion method is clearly differentiating; obtaining noticeably better results than the single counterparts, when used with the GNG algorithm. As happened for Classification Error, when the Fusion is used with the SOM, although the result obtained is better than the single model; the improvement is not as clear or significant as with GNG.



(a). Goodness of Map for the SOM-based models

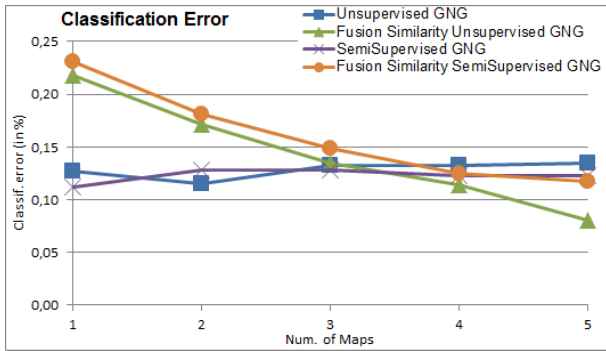


(b). Goodness of Map for the GNG-based models

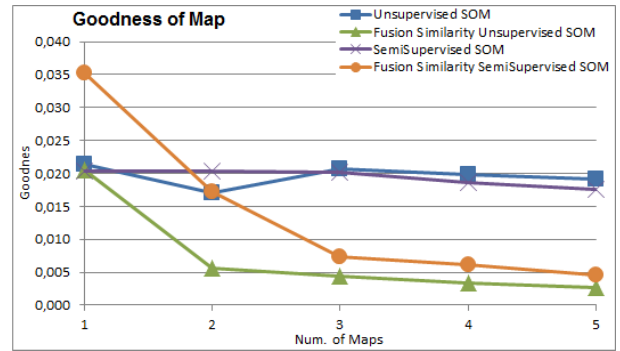


(a). Classification Error for the SOM-based models

Figure 2. Comparative of the influence of the Semi-supervised



(b). Classification Error for the GNG-based models

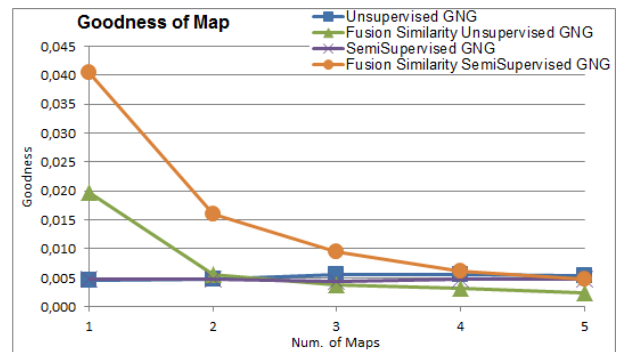


(a). Goodness of Map for the SOM-based models

Figure 3. Comparative of the influence of the Semi-supervised Learning and Ensemble Fusion algorithms on the Classification Error of the studied models when analyzing the Wine dataset.

It is interesting to note that, although the Goodness of Map measure is not directly related to the Classification Accuracy of the topology preserving maps, the improvement of the results follows a very similar pattern. This can be considered as an evidence of consistency of the results obtained, as when one aspect of the map improves or deteriorates, the other seems to behave in a similar way.

Comparing the results obtained for classification over the Wine dataset (Fig. 3) with those of the Iris dataset (Fig. 1); it becomes obvious that the second is a more complex setup. Every model has a much higher Classification Error. Under these circumstances, it is interesting to note that, although needing more composing maps added to the Fusion; the Map Fusion meta-model is able to outperform single models again. This is especially clear in the case of the use of the GNG as base classifiers.



(b). Goodness of Map for the GNG-based models

Figure 4. Comparative of the influence of the Semi-supervised Learning and Ensemble Fusion algorithms on the Goodness of Map of the studied models when analyzing the Wine dataset.

The difference between Unsupervised and Semi-Supervised in this case is also not worth mentioning for single models. Nevertheless, there is a clear difference when the Semi-supervised method is used in combination with Fusion, obtaining clearly worse results than those obtained by the Fusion trained with classical unsupervised learning. This can also be appreciated in Table 2 and Table 3 (in Appendix A) which show clear differences between results. It is evident that the results of the Fusion of Semi-Supervised trained SOMs are different from the rest of models. It is also interesting to note that in the cases of the GNG as base classifier, there are clear differences in the results between the models using the Ensemble Fusion scheme.

The Goodness of Map measure (Fig. 4) follows a similar pattern to the one of the Iris dataset. The main difference, in this case, is that the clear improvement is

obtained by the Fusion meta-algorithm but when used with the SOM as a base classifier.

This is also consistent with the results of Table 6 and Table 7; where clear differences are obtained by the SOM-based algorithms.

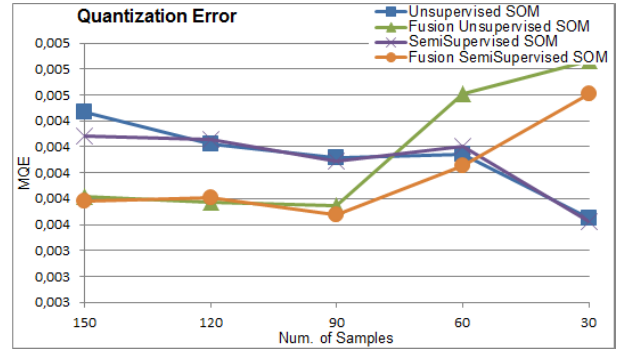
For the other three datasets, in the case of the Cancer and Echo-cardiogram datasets, the results mirror those of the Iris, being the best model the single SOM and the Fusion GNG; while in the Glass dataset, the Fusion SOM and single GNG can be considered the best models.

As with the previous observation, this seems to be due to the nature of the dataset. In the case of the Wine dataset, the samples of different classes are much more similar between them; which makes them more compactly distributed over the input space than in the Iris dataset, where three groups are separated. In this case of a continuous distribution the best shape seems to be the SOM grid. So the Fusion meta-algorithm favors this configuration, obtaining much lower errors than the single models.

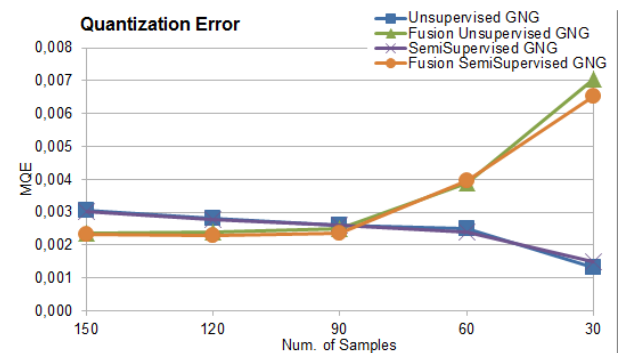
### 7.2. Experiment 2: Decreasing the size of the dataset

The second experiment consists of 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 datasets, as when using a smaller amount of data but maintaining its dimensionality, thus the training process becomes more difficult. In all the figures shown in this section, ordinate axes represent the value of the error measure, while abscissa axes represent the size of the dataset used for training. The number of composing models for the ensemble is always set to five.

The results for the Iris dataset are showed in Figure 5 and Figure 6 and some for the Wine dataset can be found in Figure 7. The complete numerical results for the other datasets analyzed are presented in Table 8, Table 9 and Table 10.



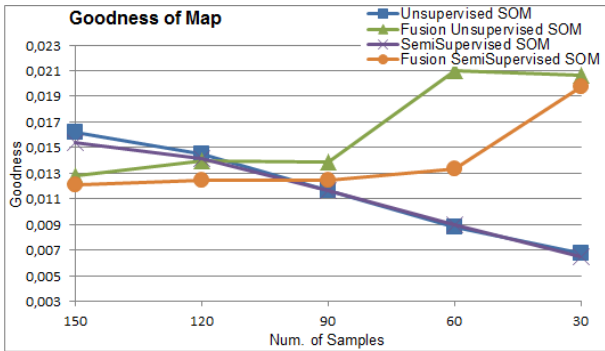
(a). Quantization Error for the SOM-based models



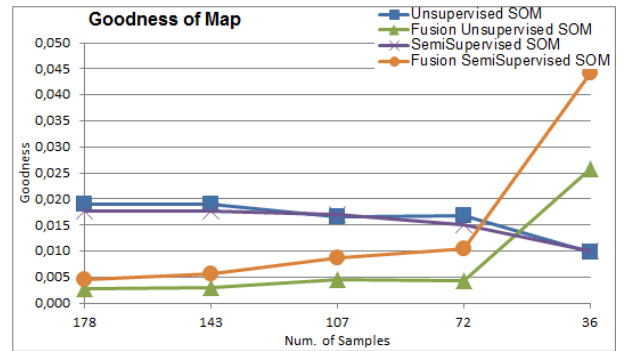
(b). Quantization Error for the GNG-based models

Figure 5. Comparative of the influence of the Semi-supervised Learning and Ensemble Fusion algorithms on the Quantization Error of the studied models when analyzing the Iris dataset.

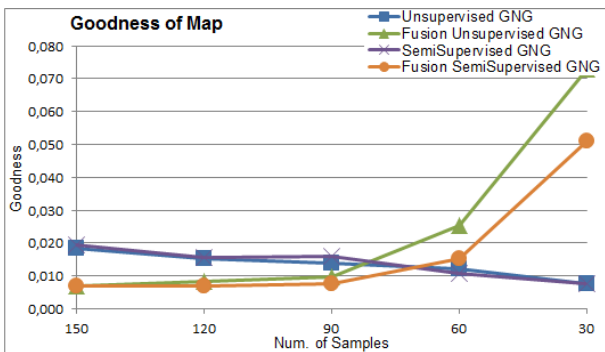
In all results in this set of experiments, the pattern obtained is similar. The Fusion method, both for unsupervised and semi-supervised learning, obtains better results than the single model until the number of samples used for the training decreases to less than the half of the initial size. At that point results start to deteriorate, becoming worse than those of the single models. This happens both for the Quantization Error (Fig. 5) and for the Goodness of Maps (Fig. 6 and Fig. 7). This is a logical consequence of low number of samples. When the dataset is too sparse, the different models of the ensemble are trained on too similar or different datasets, losing the ensemble balance between diversity and similarity that is one of the most important characteristics of this kind of meta-algorithms.



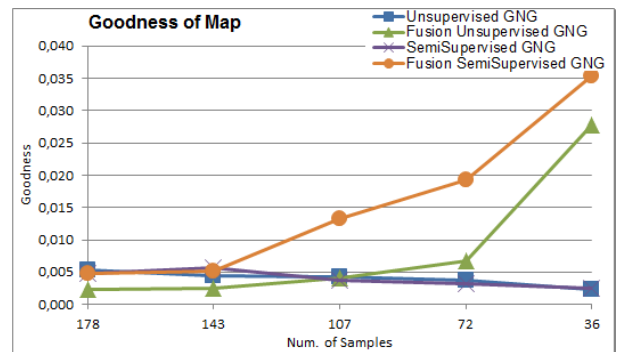
(a). Goodness of Map for the SOM-based models



(a). Goodness of Map for the SOM-based models



(b). Goodness of Map for the GNG-based models



(b). Goodness of Map for the GNG-based models

Figure 6. Comparative of the influence of the Semi-supervised Learning and Ensemble Fusion algorithms on the Goodness of Map of the studied models when analyzing the Iris dataset.

Figure 7. Comparative of the influence of the Semi-supervised Learning and Ensemble Fusion algorithms on the Goodness of Map of the studied models when analyzing the Wine dataset.

Although all measures follow that same pattern of performance degrading with the reduction of the training data, it is interesting to note that in a higher degree the GNG-based models than the SOM-based ones; for all three measures and in the five datasets presented. This is again due to the different nature of both algorithms. The fact that the SOM yields a more rigid structure in the form of a grid makes it also a more stable algorithm; as opposed to the GNG, which is a more unstructured graph and more easily influenced by that type of changes.

From the comparison of the results presented, there are several aspects that arise as general conclusions. The first is that, as happens with the majority of automated

### 7.3. Summary

Number of maps	
Trained Model	

		<p>distinguished, such as in the Iris dataset.</p> <p>In other aspects related to topology preserving models such as the Quantization Error, the use of the semi-supervised learning does not seem to have a clear benefit over the classical unsupervised learning.</p> <p>The Fusion by Voronoi Polygon Similarity algorithm proves to be interesting in cases where models are unstable. The GNG algorithm can lead to more unstable results. However, the experiments where the GNG was the base</p>		
--	--	--	--	--

		<p>the ensemble fusion algorithm and the GNG as base model obtain clearly better results than those using the SOM in datasets such as the Iris, Cancer and Echo-Cardiogram.</p> <p>It is interesting to note that when the dataset under analysis is relatively simple such as the Iris, the combination of Semi-supervised learning and ensemble fusion is the model that performs better in the majority of the cases when enough components are used. On the contrary, when a more complex</p>		
--	--	---	--	--

		<p>SOM and the GNG. The novel model called the S<sup>2</sup>-Ensemble Fusion Algorithm, involves both the bagging ensemble meta-algorithm with a fusion step and the semi-supervised learning algorithm to include unlabeled samples in the training of base components. An extensive study of the characteristics of the proposed model is included.</p> <p>From this study, it can be concluded that the use of the semi-supervised learning does not have as great an effect over</p>		
--	--	--	--	--

		<p>the ensemble.</p> <p>Nevertheless, a negative interaction between the semi-supervised learning and the ensemble fusion algorithms used within the model has been observed in the experiments, especially with more sparse datasets that include a low number of samples. This makes the improvement of the model over its single counterparts not as good as the use of the combination of ensemble fusion with unsupervised learning in some cases.</p> <p>An obvious line of future work is to investigate</p>		
--	--	---	--	--



		<p>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.</p> <p><b>Referen ces</b></p> <p>1. A. Shark</p>	
--	--	---	--

		<p><i>ion Systems</i>, vol. Vol. 7, No. 1, no. ISSN 1352-9404, pp. pp. 1–10, 2000.</p> <p>4. M. Akhand, M. M. Islam, and K. Mura se, “Compar ative study of data sampling techniqu es for construct ing neural network ensembl es,” <i>Internati onal Journal of Neural Systems</i>, vol. 19, pp. 67–89, 2009.</p> <p>5. S. Hung and H. Adeli, “Parallel backpro pagation learning algorith ms on cray y-mp8/864 superco mputer,” <i>Neuroco mputing</i>, vol. 5(6), pp. 287–</p>		
--	--	---	--	--

		<p>1997.</p> <p>8. H. Adeli and A. Karim, "Scheduling/cost optimization and neural dynamics model for construction," <i>Journal of Construction Management and Engineering</i>, ASCE, vol. Vol. 123, No. 4, pp. 450–458, 1997.</p> <p>9. H. Adeli and A. Samant, "An adaptive conjugate gradient neural network - wavelet model for traffic incident detection," <i>Computer-Aided Civil and Infrastructure Engineering</i>, vol. Vol. 15, No. 4, pp. 251–</p>		
--	--	--	--	--

		<p>12. A. Khas hman and B. Seker oglu, “Docum ent image binarisat ion using a supervis ed neural network, ”  <i>Internati onal Journal of Neural Systems</i>, vol. Vol. 18, No. 5, pp. 405– 418, 2008.</p> <p>13. J. E. Menke and T. R. Martinez , “Improvi ng supervis ed learning by adapting the problem to the learner,” <i>Int. J. Neural Syst.</i>, vol. 19, no. 1, pp. 1–9, 2009.</p> <p>14. A. Panak kat and H. Adeli, “Recurr ent neural network</p>		
--	--	---	--	--

		<p>and H. Adeli, "A new supervis ed learning algorith m for multiple spiking neural networks with applicati on in epilepsy and seizure detection ," <i>Neural Netw.</i>, vol. 22, no. 10, pp. 1419– 1431, 2009.</p> <p>17. M. Ahm adlou and H. Adeli, "Enhanc ed probabili stic neural network with local decision circles: A robust classifier ," <i>Integrate d Compute r-Aided Engineer ing</i>, vol. 17:3, pp. 197– 210, 2010.</p> <p>18. W. Graf, S. Freita</p>		
--	--	--	--	--

		<p>algorithm for radial basis function neural network with application to chemical oxygen demand modeling.”</p> <p><i>International Journal of Neural Systems</i>, vol. 20:1, pp. 63–74, 2010.</p> <p>21. S. Schliebs, N. Kasabv, and M. Defoin-Platel, “On the probabilistic optimization of spiking neural networks,” <i>International Journal of Neural Systems</i>, vol. 20:6, pp. 481–500, 2010.</p> <p>22. G. Giacinto and F. Roli, “Design of effective</p>		
--	--	--	--	--

		<p>“Boosting unsupervised competitive learning ensembles,” in <i>17th International Conference in Artificial Neural Networks (ICANN 2007)</i>, ser. Lecture Notes in Computer Science, vol. Porto, Portugal. Springer Berlin / Heidelberg, 2007, pp. 339–348.</p> <p>25. B. Barua and E. Corchado, “A weighted voting summarization of SOM ensembles,” <i>Data Mining and Knowledge Discovery</i>, vol. 21 (3), pp. 398–426, 2010,</p>		
--	--	--	--	--

		<p>28. W. Chen and C. Zhang, "A hybrid framework for protein sequence clustering and classification using functional signature motif information," <i>Integrated Computer-Aided Engineering</i>, vol. 16:4, pp. 353–365, 2009.</p> <p>29. A. Rodriguez and J. Reggia, "A distributed learning algorithm for particle systems," <i>Integrated Computer-Aided Engineering</i>, vol. Vol. 16, No. 1, pp. 1–20, 2009.</p> <p>30. L. Carro-Calvo, S. Salce</p>		
--	--	---	--	--



		<p><i>onal Journal of Neural Systems</i>, vol. 21:1, pp. 17–29, 2011.</p> <p>32. U. Reute r, “A fuzzy approach for modellin g non- stochasti c heteroge neous data in engineer ing based on cluster analysis, ” <i>Integrate d Compute r-Aided Engineer ing</i>, vol. 18:3, 2011.</p> <p>33. A. Fred and A. Jain, “Combin ing multiple clusterin gs using evidence accumul ation,” <i>Pattern Analysis and Machine Intellige nce, IEEE Transact ions on</i>, vol. 27, no. 6,</p>		
--	--	--	--	--

		<p>e solution, ”</p> <p><i>Artificial Intelligence in Medicine</i>, vol. 20, pp. 217– 225, 1999.</p> <p>36. R. Polik ar, “Ensemb le based systems in decision making,” <i>IEEE Circuits and Systems Magazin e</i>, vol. 6, no. 3, pp. 21– 45, 2006.</p> <p>37. H. Adeli and S. Hung, <i>Machine Learning - Neural Network s, Genetic Algorith ms, and Fuzzy Systems</i>. New York, USA,: Wiley &amp; Sons, 1995.</p> <p>38. G. D. A. Barreto and A. F. R. Araújo, “Unsupe rvised learning</p>		
--	--	---	--	--

		<p>pp. 444–445, 1974.</p> <p>42. E. Lopez-Rubio and J. Ortiz-de Lazaño-Lobato, “Dynamic competitive probabilistic principal components analysis,” <i>International Journal of Neural Systems</i>, vol. 19:2, pp. 91–103, 2009.</p> <p>43. T. Kohonen, “The self-organizing map,” <i>Neurocomputing</i>, vol. 21, no. 1-3, pp. 1–6, 1998.</p> <p>44. A. Fatehi and K. Abe, “Flexible structure multiple modeling using irregular self-organizing maps neural</p>		
--	--	---	--	--

		<p>”  <i>International Journal of Neural Systems</i>,  vol.  20:5, pp.  389–  403,  2010.</p> <p>47. E. Lopez-Rubio, R. Luque-Baena, and E. Dominguez, “Foreground detection in video sequences with probabilistic self-organizing maps,” <i>International Journal of Neural Systems</i>, vol. 21:3, pp. 225–246, 2011.</p> <p>48. H. Adeli and H. Park, “Counterpropagation neural network in structural engineering,” <i>Journal of Structural</i></p>		
--	--	--	--	--

		<p>o, and J. de Go y, "Distrib uted intrusion detection system for wireless sensor networks based on a reputatio n system coupled with kernel self-organizin g maps," <i>Integrate d Computer-Aided Engineer ing</i>, vol. 17:2, pp. 87-102, 2010.</p> <p>52. D. Boto-Giralda, F. Díaz-Pernas, D. Gonz ález-Ortega, J. Díez-Higuera, M. Antón-Rodríguez, and M. Martínez-Zarzuela , "Wavelet-based denoisin g for traffic volume time series forecasti</p>		
--	--	--	--	--

		<p>principle of neural associative memory,"</p> <p><i>Neuroscience</i>, vol. 2, no. 6, pp. 1065–1076, 1977.</p> <p>55. T. Kohonen, "An introduction to neural computing," <i>Neural Networks</i>, vol. 1, no. 1, pp. 3–16, 1988.</p> <p>56. T. Kohonen, <i>Self-Organizing Maps</i>. Berlin, Germany: Springer, 1995, vol. 30.</p> <p>57. B. Fritzke, "Unsupervised clustering with growing cell structures," in <i>IJCNN'91 - Seattle International Joint Conference</i></p>		
--	--	---	--	--

		<p>comparison of quality measures for self-organizing maps,” in <i>Fifth Workshop on Data Analysis (WDA'04)</i>, J. Paralic, G. Polzlbauer, and R. Andreas, Eds. Elfa Academic Press, 2004, pp. 67–82.</p> <p>61. S. M. Zaki and H. Yin, “A semi-supervised learning algorithm for growing neural gas in face recognition,” <i>Journal of Mathematical Modelling and Algorithms</i>, vol. 7, pp. 425–435, 2008. [Online].</p>		
--	--	---	--	--

		<p>C. v. d. Malsburg, W. v. Seelen, J. C. Vorbruggen, and B. Sendhoff, Eds., vol. Bochum, Germany. Berlin: Springer, 1996, pp. 809–814.</p> <p>64. R. Dara, S. C. Kremer, and D. A. Stacey, "Clustering unlabelled data with SOMs improves classification of labelled real-world data," in <i>Proc. IEEE World Congress on Computational Intelligence</i>, May 2002, pp. 2237–2242.</p> <p>65. A. Bouc hachia, "Learnin g with partly</p>		
--	--	---	--	--



		<p>supervised learning for web text clustering,” in <i>IC-AI</i>, 2006, pp. 243–249.</p> <p>68. D. Angluin, “Queries and concept learning,” <i>Machine Learning</i>, vol. 2, p. 319–342, 1988.</p> <p>69. Y. Freund, H. Seung, E. Shamir, and N. Tishby, “Selective sampling using the query by committee algorithm,” <i>Machine Learning</i>, vol. 28, pp. 133–168, 1997.</p> <p>70. S. Ji and L. Carin, “Cost-sensitive feature acquisition and classification,” <i>Pattern</i></p>		
--	--	---	--	--

		<p>1996.</p> <p>74. D. Ruta and B. Gabrys, "Classifier selection for majority voting," <i>Information Fusion</i>, vol. 6, no. 1, pp. 63–81, 2005.</p> <p>75. L. I. Kuncheva, <i>Combining Pattern Classifiers: Methods and Algorithms</i>. Wiley-Interscience, 2004, vol. 47, no. 4.</p> <p>76. B. Barua and E. Corchado, "A novel ensemble of scale-invariant feature maps," in <i>CORES'09, International Conference on Computer</i></p>		
--	--	--	--	--

		<p>ion artificial neural networks ,” <i>Pattern Recognit ion Letters</i>, vol. 32, no. 7, pp. 962– 972, 2011.</p> <p>79. F. Auren hammer and R. Klein, “Voronoi diagrams ,” in <i>Handbo ok of Computa tional Geometr y</i>. Amsterd am, Netherla nds: North- Holland, 2000, no. 5, pp. 201– 290.</p> <p>80. A. Frank and A. Asun cion, “UCI machine learning repositor y,” 2010. [Online]. Availabl e: <a href="http://archive.ics.uci.edu/ml">http://- archive.i cs.uci.ed u/ml</a></p> <p><b>Appendix A.</b></p>		
--	--	---	--	--

		<p>datasets in the study.</p> <p>For each dataset and each number of maps used, the lowest error for SOM-based and for GNG-based combinations are highlighted.</p> <p>Tables 2 and 3 present the results of applying the student's t-test to the results showed in Table 1 (classification error); while Tables 6 and 7 represent the same calculation for results of Table 5 (goodness of maps). The results that represent a significant statistical difference (value lower than 0.05) are highlighted.</p>		
--	--	--	--	--

n		
---	--	--

Taking into account the results obtained for the Classification Error measure, it can be concluded that the Semi-supervised algorithms have an effect on the way the topology preserving maps are trained, despite not being the models specifically designed for that purpose. Also the improvement is noticeable in cases in which classes are more clearly distinguished, such as in the Iris dataset.

In other aspects related to topology preserving models such as the Quantization Error, the use of the semi-supervised learning does not seem to have a clear benefit over the classical unsupervised learning.

The Fusion by Voronoi Polygon Similarity algorithm proves to be interesting in cases where models are unstable. The GNG algorithm can lead to more unstable results. However, the experiments where the GNG was the base classifier show higher improvements for the ensemble fusion models. It is important to note that although the fusion algorithm adds certain stability to the models, it is not able to fully counter the added instability when the dataset decreases. In those particular cases, simple algorithms seem to more stable.

Another insightful observation is that, as the fusion model used favors the GNG algorithm rather than the SOM due to its less constricted way of adapting, the models including the ensemble fusion algorithm and the GNG as base model obtain clearly better results than those using the SOM in datasets such as the Iris, Cancer and Echo-Cardiogram.

It is interesting to note that when the dataset under analysis is relatively simple such as the Iris, the combination of Semi-supervised learning and ensemble fusion is the model that performs better in the majority of the cases when enough components are used. On the contrary, when a more complex dataset such as the Wine; the best combination seems to be the unsupervised learning with ensemble fusion, when enough maps or not that much instability is added. This indicates a negative interaction between both models in these cases that will require a deeper study in the future.

## 8. Conclusions

This research presents a novel combination of algorithms that can be applied to improve the characteristics of two well-known topology preserving models: the SOM and the GNG. The novel model called the  $S^2$ -Ensemble Fusion Algorithm, involves both the bagging ensemble meta-algorithm with a fusion step and the semi-supervised learning algorithm to include unlabeled samples in the training of base components. An extensive study of the characteristics of the proposed model is included.

From this study, it can be concluded that the use of the semi-supervised learning does not have as great an effect over topology preserving models as the use of ensembles. The model proposed proves its usefulness when enough base components are used to construct the ensemble and the size of the dataset used for its training is large enough to ensure the stability of the training. It is shown that it improves the analysis performed using GNG base models over not excessively sparse datasets, such as the Iris, Cancer or Echo-Cardiogram for all the analyzed measures; provided that enough base models are added to the ensemble.

Nevertheless, a negative interaction between the semi-supervised learning and the ensemble fusion algorithms used within the model has been observed in the experiments, especially with more sparse datasets that include a low number of samples. This makes the improvement of the model over its single counterparts not as good as the use of the combination of ensemble fusion with unsupervised learning in some cases.

An obvious line of future work is to investigate the instability observed in sparse datasets. Also the test of the model with other type of topology preserving algorithms is considered as another line of work, as well as the devising of fusion methods more suitable for more rigid algorithms. Finally, applications of the novel models to real-life problems and its inclusion as a part of more complex hybrid systems for different tasks is also intended.

## 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.

## References

1. A. Sharkey and N. Sharkey, "Diversity, selection and ensembles of artificial neural nets," in *Third International Conference on Neural Networks and their Applications*, 1997, pp. 205–212.
2. A. Sharkey and N. Sharkey, "Combining diverse neural nets," *Knowledge Engineering Review*, vol. 12, 3, pp. 1–17, 1997.
3. D. Ruta and B. Gabrys, "An overview of classifier fusion methods," *Computing and Information Systems*, vol. Vol. 7, No. 1, no. ISSN 1352-9404, pp. pp. 1–10, 2000.
4. M. Akhand, M. M. Islam, and K. Murase, "Comparative study of data sampling techniques for constructing neural network ensembles," *International Journal of Neural Systems*, vol. 19, pp. 67–89, 2009.
5. S. Hung and H. Adeli, "Parallel backpropagation learning algorithms on cray y-mp8/864 supercomputer," *Neurocomputing*, vol. 5(6), pp. 287–302, 1993.
6. H. Park and H. Adeli, "Distributed neural dynamics algorithms for optimization of large steel structures," *Journal of Structural Engineering, ASCE*, vol. Vol. 123, No. 7, pp. 880–888, 1997.
7. H. Adeli and A. Karim, "Neural network model for optimization of cold-formed steel beams," *Journal of Structural Engineering, ASCE*, vol. Vol. 123, No. 11, pp. 1535–1543, 1997.
8. H. Adeli and A. Karim, "Scheduling/cost optimization and neural dynamics model for construction," *Journal of Construction Management and Engineering, ASCE*, vol. Vol. 123, No. 4, pp. 450–458, 1997.
9. H. Adeli and A. Samant, "An adaptive conjugate gradient neural network - wavelet model for traffic incident detection," *Computer-Aided Civil and Infrastructure Engineering*, vol. Vol. 15, No. 4, pp. 251–260, 2000.
10. A. Senouci and H. Adeli, "Resource scheduling using neural dynamics model of adeli and park," *Journal of Construction Engineering and Management, ASCE*, vol. Vol. 127, No. 1, pp. 28–34, 2001.
11. A. Panakkat and H. Adeli, "Neural network models for earthquake magnitude prediction using multiple seismicity indicators," *International Journal of Neural Systems*, vol. 17 (1), pp. 13–33, 2007.
12. A. Khashman and B. Sekeroglu, "Document image binarisation using a supervised neural network," *International Journal of Neural Systems*, vol. Vol. 18, No. 5, pp. 405–418, 2008.
13. J. E. Menke and T. R. Martinez, "Improving supervised learning by adapting the problem to the learner," *Int. J. Neural Syst.*, vol. 19, no. 1, pp. 1–9, 2009.
14. A. Panakkat and H. Adeli, "Recurrent neural network for approximate earthquake time and location prediction using multiple seismicity indicators," *Computer-Aided Civil and Infrastructure Engineering*, vol. 24:4, pp. 280–292, 2009.
15. R. Lin, J. Liang-Te Chiu, H. Dai, R. Tsai, M. Day, and W. Hsu, "A supervised learning approach to biological question answering," *Integrated Computer-Aided Engineering*, vol. 16:3, pp. 271–281, 2009.
16. S. Ghosh-Dastidar and H. Adeli, "A new supervised learning algorithm for multiple spiking neural networks with application in epilepsy and seizure detection," *Neural Netw.*, vol. 22, no. 10, pp. 1419–1431, 2009.
17. M. Ahmadlou and H. Adeli, "Enhanced probabilistic neural network with local decision circles: A robust classifier," *Integrated Computer-Aided Engineering*, vol. 17:3, pp. 197–210, 2010.
18. W. Graf, S. Freitag, M. Kaliske, and J. Sickert, "Recurrent neural networks for uncertain time-dependent structural behavior," *Computer-Aided Civil and Infrastructure Engineering*, vol. 25:5, pp. 322–333, 2010.
19. U. Reuter and B. Moeller, "Artificial neural networks for forecasting of fuzzy time series," *Computer-Aided Civil and Infrastructure Engineering*, vol. 25:5, pp. 363–374, 2010.
20. Q. Junfei and H. Honggui, "A repair algorithm for radial basis function neural network with application to chemical oxygen demand modeling," *International Journal of Neural Systems*, vol. 20:1, pp. 63–74, 2010.
21. S. Schliebs, N. Kasabv, and M. Defoin-Platel, "On the probabilistic optimization of spiking neural networks," *International Journal of Neural Systems*, vol. 20:6, pp. 481–500, 2010.
22. G. Giacinto and F. Roli, "Design of effective neural network ensembles for image classification purposes," *Image and Vision Computing*, vol. Volume 19, Issues 9-10, pp. 699–707, 2001.
23. M. Bundzel and P. Sincik, "Ensembling classifiers using unsupervised learning," in *Artificial Intelligence and Soft Computing - ICAISC 2008*, ser. Lecture Notes in Computer Science, vol. 5097/2008, 2008, pp. 513–521.
24. E. Corchado, B. Baruque, and H. Yin, "Boosting unsupervised competitive learning ensembles," in *17th International Conference in Artificial Neural Networks (ICANN 2007)*, ser. Lecture Notes in Computer Science, vol. Porto, Portugal. Springer Berlin / Heidelberg, 2007, pp. 339–348.
25. B. Baruque and E. Corchado, "A weighted voting summarization of SOM ensembles," *Data Mining and Knowledge Discovery*, vol. 21 (3), pp. 398–426, 2010, 10.1007/s10618-009-0160-3.
26. E. Corchado and B. Baruque, "Wevos-visom: An ensemble summarization algorithm for enhanced data

- visualization,” *Neurocomputing*, vol. In Press, Corrected Proof, pp. –, 2011.
27. S. Ghosh-Dastidar and H. Adeli, “Wavelet-clustering-neural network model for freeway incident detection,” *Computer-Aided Civil and Infrastructure Engineering*, vol. Vol. 18, No. 5, pp. 325–338, 2003.
  28. W. Chen and C. Zhang, “A hybrid framework for protein sequences clustering and classification using functional signature motif information,” *Integrated Computer-Aided Engineering*, vol. 16:4, pp. 353–365, 2009.
  29. A. Rodriguez and J. Reggia, “A distributed learning algorithm for particle systems,” *Integrated Computer-Aided Engineering*, vol. Vol. 16, No. 1, pp. 1–20, 2009.
  30. L. Carro-Calvo, S. Salcedo-Sanz, R. Gil-Pita, A. Portilla-Figueras, and M. Rosa-Zurera, “An evolutive multiclass algorithm for automatic classification of high range resolution radar targets,” *Integrated Computer-Aided Engineering*, vol. 16:1, pp. 51–60, 2009.
  31. R. Cruz-Barbosa and A. Vellido, “Semi-supervised analysis of human brain tumour from partially labeled MRS information using manifold learning models,” *International Journal of Neural Systems*, vol. 21:1, pp. 17–29, 2011.
  32. U. Reuter, “A fuzzy approach for modelling non-stochastic heterogeneous data in engineering based on cluster analysis,” *Integrated Computer-Aided Engineering*, vol. 18:3, 2011.
  33. A. Fred and A. Jain, “Combining multiple clusterings using evidence accumulation,” *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 27, no. 6, pp. 835 – 850, jun 2005.
  34. S. Porras, A. Alonso, B. Baruque, H. Yin, E. Corchado, and J. Rovira, “Improving the growing neural gas algorithm with ensembles,” in *10th International Conference on Computational and Mathematical Methods in Science and Engineering (CMMSE 2010)*, 2010.
  35. P. Cunningham, J. Carney, and S. Jacob, “Stability problems with artificial neural networks and the ensemble solution,” *Artificial Intelligence in Medicine*, vol. 20, pp. 217–225, 1999.
  36. R. Polikar, “Ensemble based systems in decision making,” *IEEE Circuits and Systems Magazine*, vol. 6, no. 3, pp. 21–45, 2006.
  37. H. Adeli and S. Hung, *Machine Learning - Neural Networks, Genetic Algorithms, and Fuzzy Systems*. New York, USA,: Wiley & Sons, 1995.
  38. G. D. A. Barreto and A. F. R. Araújo, “Unsupervised learning and temporal context to recall complex robot trajectories,” *Int. J. Neural Syst.*, vol. 11, no. 1, pp. 11–22, 2001.
  39. M. H. Hassoun, *Fundamentals of Artificial Neural Networks*. MIT Press, 1995.
  40. D. O. Hebb, *The Organization of Behavior: A Neuropsychological Theory*. Lawrence Erlbaum Associates, 1949.
  41. T. Kohonen, “Adaptive associative memory principle,” *Ieee Transactions on Computers*, vol. C 23, no. 4, pp. 444–445, 1974.
  42. E. Lopez-Rubio and J. Ortiz-de Lazcano-Lobato, “Dynamic competitive probabilistic principal components analysis,” *International Journal of Neural Systems*, vol. 19:2, pp. 91–103, 2009.
  43. T. Kohonen, “The self-organizing map,” *Neurocomputing*, vol. 21, no. 1-3, pp. 1–6, 1998.
  44. A. Fatehi and K. Abe, “Flexible structure multiple modeling using irregular self-organizing maps neural network,” *International Journal of Neural Systems*, vol. 18(3), pp. 233–256, 2008.
  45. H. Ni and H. Yin, “Self-organizing mixture autoregressive model for non-stationary time series prediction,” *International Journal of Neural Systems*, vol. Vol. 18, No. 6, 2008.
  46. N. Wang, M. Joo, X. Meng, and X. Li, “An online self-organizing scheme for parsimonious and accurate fuzzy neural networks,” *International Journal of Neural Systems*, vol. 20:5, pp. 389–403, 2010.
  47. E. Lopez-Rubio, R. Luque-Baena, and E. Dominguez, “Foreground detection in video sequences with probabilistic self-organizing maps,” *International Journal of Neural Systems*, vol. 21:3, pp. 225–246, 2011.
  48. H. Adeli and H. Park, “Counter propagation neural network in structural engineering,” *Journal of Structural Engineering, ASCE*, vol. Vol. 121, No. 8, pp. 1205–1212, 1995.
  49. H. Yin, “ViSOM - a novel method for multivariate data projection and structure visualization,” *Neural Networks, IEEE Transactions on*, vol. 13, no. 1, pp. 237–243, 2002.
  50. H. Yin, “Data visualisation and manifold mapping using the ViSOM,” *Neural Networks*, vol. 15, no. 8-9, pp. 1005–1016, 2002.
  51. Z. Bankovic, J. Moya, A. Araujo, D. Fraga, J. Vallejo, and J. de Goy, “Distributed intrusion detection system for wireless sensor networks based on a reputation system coupled with kernel self-organizing maps,” *Integrated Computer-Aided Engineering*, vol. 17:2, pp. 87–102, 2010.
  52. D. Boto-Giralda, F. Díaz-Pernas, D. González-Ortega, J. Díez-Higuera, M. Antón-Rodríguez, and M. Martínez-Zarzuela, “Wavelet-based denoising for traffic volume time series forecasting with self-organizing neural networks,” *Computer-Aided Civil and Infrastructure Engineering*, vol. 25:7, pp. 530–545, 2010.
  53. M.-L. Nichols, E. and N. Siddique, “Case study on self-organizing spiking neural networks for robot navigation,” *International Journal of Neural Systems*, vol. 20:6, pp. 501–508, 2010.
  54. T. Kohonen, P. Lehtio, J. Rovamo, J. Hyvarinen, K. Bry, and L. Vainio, “A principle of neural associative memory,” *Neuroscience*, vol. 2, no. 6, pp. 1065–1076, 1977.
  55. T. Kohonen, “An introduction to neural computing,” *Neural Networks*, vol. 1, no. 1, pp. 3–16, 1988.

56. T. Kohonen, *Self-Organizing Maps*. Berlin, Germany: Springer, 1995, vol. 30.
57. B. Fritzke, "Unsupervised clustering with growing cell structures," in *IJCNN'91 - Seattle International Joint Conference on Neural Networks*, vol. 2, Jul 1991, pp. 531–536.
58. T. M. Martinetz and K. J. Schulten, "A neural-gas network learns topologies," *Artificial Neural Networks*, pp. 397–402, 1991.
59. T. Martinetz, "Competitive hebbian learning rule forms perfectly topology preserving maps," in *Proceeding of the Int. Conf. on Artificial Neural Networks (ICANN 93)*, 1993, pp. 426–438.
60. G. Pözlbauer, "Survey and comparison of quality measures for self-organizing maps," in *Fifth Workshop on Data Analysis (WDA'04)*, J. Paralic, G. Polzlbauer, and R. Andreas, Eds. Elfa Academic Press, 2004, pp. 67–82.
61. S. M. Zaki and H. Yin, "A semi-supervised learning algorithm for growing neural gas in face recognition," *Journal of Mathematical Modelling and Algorithms*, vol. 7, pp. 425–435, 2008. [Online]. Available: <http://www.springerlink.com/content/10868425048x4064>
62. A. Georgakis, H. Li, and M. Gordan, "An ensemble of SOM networks for document organization and retrieval," in *International Conference on Adaptive Knowledge Representation and Reasoning (AKRR'05)*, 2005, pp. 6–141.
63. S. Kaski and K. Lagus, "Comparing self-organizing maps," in *Lecture Notes in Computer Science*, C. v. d. Malsburg, W. v. Seelen, J. C. Vorbruggen, and B. Sendhoff, Eds., vol. Bochum, Germany. Berlin: Springer, 1996, pp. 809–814.
64. R. Dara, S. C. Kremer, and D. A. Stacey, "Clustering unlabelled data with SOMs improves classification of labelled real-world data," in *Proc. IEEE World Congress on Computational Intelligence*, May 2002, pp. 2237–2242.
65. A. Bouchachia, "Learning with partly labeled data," *Neural Computing Applications*, vol. 16, pp. 267–293, 2006.
66. A. Hsu and S. K. Halgamuge, "Class structure visualization with semi-supervised growing self-organizing maps," *Neurocomputing*, vol. 71, no. 16-18, pp. 3124 – 3130, 2008, advances in Neural Information Processing (ICONIP 2006) / Brazilian Symposium on Neural Networks (SBRN 2006).
67. B. Yang, W. Song, and Z. Xu, "Semi-supervised learning for web text clustering," in *IC-AI*, 2006, pp. 243–249.
68. D. Angluin, "Queries and concept learning," *Machine Learning*, vol. 2, p. 319–342, 1988.
69. Y. Freund, H. Seung, E. Shamir, and N. Tishby, "Selective sampling using the query by committee algorithm," *Machine Learning*, vol. 28, pp. 133–168, 1997.
70. S. Ji and L. Carin, "Cost-sensitive feature acquisition and classification," *Pattern Recognition*, vol. 40, p. 1474–1485, 2007.
71. M. Saar-Tsechansky, P. Melville, and F. Provost, "Active feature-value acquisition," *Management Science*, vol. 55(4), p. 664–684, 2009.
72. B. Settles, "Active learning literature survey," University of Wisconsin–Madison, Computer Sciences Technical Report 1648, 2009.
73. L. Breiman, "Bagging predictors," *Machine Learning*, vol. 24, no. 2, pp. 123–140, 1996.
74. D. Ruta and B. Gabrys, "Classifier selection for majority voting," *Information Fusion*, vol. 6, no. 1, pp. 63–81, 2005.
75. L. I. Kuncheva, *Combining Pattern Classifiers: Methods and Algorithms*. Wiley-Interscience, 2004, vol. 47, no. 4.
76. B. Baruque and E. Corchado, "A novel ensemble of scale-invariant feature maps," in *CORES'09, International Conference on Computer Recognition Systems*, ser. Advances in Soft Computing, vol. 57. Springer, 2009, pp. 265–273.
77. C. Saavedra, R. Salas, S. Moreno, and H. Allende, "Fusion of self-organizing maps," in *9th International Work-Conference on Artificial Neural Networks (IWANN 2007)*, 2007, pp. 227–234.
78. R. Salas, C. Saavedra, H. Allende, and C. Moraga, "Machine fusion to enhance the topology preservation of vector quantization artificial neural networks," *Pattern Recognition Letters*, vol. 32, no. 7, pp. 962–972, 2011.
79. F. Aurenhammer and R. Klein, "Voronoi diagrams," in *Handbook of Computational Geometry*. Amsterdam, Netherlands: North-Holland, 2000, no. 5, pp. 201–290.
80. A. Frank and A. Asuncion, "UCI machine learning repository," 2010. [Online]. Available: <http://archive.ics.uci.edu/ml>



**Appendix A.**

**A.1. Experiment 1**

The first experiment consists of using the complete dataset to train ensembles of unsupervised or semi-supervised models; increasing the number of maps used to construct the ensemble, to assess the effect of the modification in the number of components of the ensemble. Tables 2, 5 and 6 present, respectively, the numerical results for the classification error, the

quantization error and the goodness of map for the three additional datasets in the study.

For each dataset and each number of maps used, the lowest error for SOM-based and for GNG-based combinations are highlighted.

Tables 2 and 3 present the results of applying the student's t-test to the results showed in Table 1 (classification error); while Tables 6 and 7 represent the same calculation for results of Table 5 (goodness of maps). The results that represent a significant statistical difference (value lower than 0.05) are highlighted.

Table 1. Classification Error

<b>Trained Model</b>	<b>Number of maps</b>

o or it e  n e e  re  ot e  or  e n is  in  re y e in is  er  to  h e e e e	
---	--

ot  
o  
ar  
er  
al  
e  
  
y  
  
e  
es  
e  
G  
o  
  
s  
e  
is  
e  
  
er  
n  
e  
  
is  
o  
at  
e  
  
in  
o  
S,  
le  
y

--	--

ne		
ne		
ge		
o		
.		
r		
n		
as		
on		
d		
ne		
n		
M		
ts		
of		
ls		
ne		
as		
el		
n		
g		
in		
ne		
er		
D-		

n.  
is  
at  
e  
is  
h  
s,  
n  
i-  
d  
e  
at  
e  
of  
es  
s  
n  
y,  
a  
h  
e;  
st  
n  
e  
e  
g  
ot  
h  
is  
is

--	--

a  
ls  
se  
at  
re  
er  
e  
  
a  
n  
  
be  
to  
e  
i  
o  
n  
  
e  
d  
G.  
el  
  
e  
  
e  
  
a  
p

--	--

te		
o		
n		
g		
se		
s.		
te		
i		
te		
is		
it		
be		
se		
i-		
ot		
as		
n		
er		
as		
of		
el		
ts		
se		
s		
to		
ze		

e d ts is  co e of g. n it  is  G ls ot  e er o- n e  h ls to  n a  e  d  n el n	
---	--



in s ly re at a er s. es n e er le s d of n le h e in . e re to in st el er of er of as	
---	--

e of  or d  s el o  ts as of  or o  as  d  of h of	
---	--

ta la n. rs o k  e of s le  e e )  y h of d  k d k  it  n  l of l  in  ti  e on  k d	
---	--

<p><i>ti</i></p> <p>i-</p> <p>k</p> <p>d</p> <p>k</p> <p>n</p> <p><i>d</i></p> <p><i>er</i></p> <p>,</p> <p>2,</p> <p>-</p> <p>a</p> <p>y</p> <p>an</p> <p>of</p> <p>er</p> <p>s,</p> <p><i>ti</i></p> <p><i>d</i></p> <p><i>ut</i></p> <p>,</p> <p>ol.</p> <p>l,</p> <p>N</p> <p>p.</p> <p>a</p> <p>a</p> <p>ar</p> <p>of</p>		
---	--	--

<p>g u or ct</p> <p>k l</p> <p><i>ti</i></p> <p>,</p> <p>-</p> <p>g</p> <p>li, el o n g</p> <p>on y- 4 ” o g, ), -</p> <p>li, o</p> <p>c</p> <p>or a of</p> <p>e</p>	
--	--

<p>r er bl. o. p. li l k a of e r er bl. o. p. li ul t a d c el ct r ac z</p>	
---	--

<i>d</i> <i>er</i>  bl. o. p.  li  a n e at  t  k et  t on  te  <i>d</i> <i>u</i>  <i>er</i> bl. 5, 4, -  o d li, r  i g  c el li    <i>ac</i>  <i>er</i>		
--	--	--

<p><i>d</i> <i>e</i></p> <p>bl. o. p.</p> <p>ak nd li, l k</p> <p>a</p> <p>i</p> <p>o g e it</p> <p>or</p> <p><i>ti</i></p> <p>, 7 p.</p> <p>s</p> <p>er</p> <p>n</p> <p>at g</p> <p>s al k, <i>ti</i></p>		
--	--	--



;.  
l.  
o.  
p.  
  
ez  
vi  
s  
g  
g  
n  
ne  
”  
J.  
  
9,  
k  
d  
li,  
e  
al  
k  
  
a  
ne  
  
n  
o  
g  
e  
it  
  
or  
te  
r  
d  
u  
er

--	--

ol. p.		
g- l,		
,		
,		
s		
g ch		
c		
n n		
te		
te		
er		
ol. p.		
ch		
r		
li, w s		
g		
Or e		
ks		
li in /		
on		

--	--	--

<p>u er ol. p. e nd ll i al ks i of te r d u er ol. p. ei g A Or n k i co al n ti</p>	
---	--

--	--	--

d id k  b  er g v  g, in <i>al</i> <i>e</i> <i>d</i>  ti -  n te  ,  D -  h q d  n v it  g l in  ti  <i>e</i> <i>in</i> <i>al</i>  k  J		
--	--	--

<p>in te , l. r / be 7, - q id h A ed ri of l  d er ol. , - / -  h id q -  l ri  or e ta a</p>	
--	--

<p>o 3, in e f, h r li, e n l k y t on te d u er l. 8, 5, - n n A o or ce n d e n re t</p>	
--	--



<p>te te er bl. p.  ri ad i A t  g or  s, te te er bl. 5, 1, —  o e z,  il s, a e y as  or ti e of</p>	
--	--

<p>o ar , te te er bl. p. - a i s s y at g d g ” ti y , p. e A ch n - ti e</p>	
--	--

in r on s, te te er ol.  n e n g e l  s e e  ct n,  5 0,  as as q  h d ra  vi ne g	
--	--

h l in  ti  n ta  n  s  er  )  ni  e d o, t  n h l  ks ne l  a,  al e in n  -  k  b ed s  n ”		
---	--	--

s r n 5, — li g, e g al k  h d  .  &  r  e  g  al  ll x  ri t.  —  n, n of		
---	--	--

<p>al k T</p> <p>z of or A s i</p> <p>c n at D. o</p> <p>iv</p> <p>li</p> <p>y e ee et on te</p> <p>bl. 3,</p> <p>—</p> <p>cz</p> <p>-</p> <p>o , ni</p> <p>it</p> <p>li</p> <p>al ne</p> <p>s,</p>		
---	--	--

<p>ti p. o i ” o 3, 3, 6, h id l e e n g r i os k, ti i</p>	
---	--

r		
n-		
r		
ne		
o		
ti		
,		
bl.		
o.		
g		
o,		
g		
id		
i		
o		
e		
ks		
ti		
,		
p.		
z		
i		
a,		
i		
o		



<p>on eo ce ch li f- i</p> <p><i>ti</i></p> <p>,</p> <p>p.</p> <p>li</p> <p>, er at</p> <p>k a r</p> <p>r</p> <p><i>er</i></p> <p>l. o. p.</p> <p>M el</p> <p>ri ta o d e</p>	
---	--

a k E ct n, L		
a d d g ne t, al k		
9,		
k a, j a j d o o n on		
s ks on		
o m l o		

<p>n ” te  te  er ol. p.  - - - z  a, ó  u d tí  a  e  n or  i h  i  ks  te  d u  er ol. p.  s,</p>	
---	--

d i  on  i  ks ot i  <i>ti</i>  e, p.  o io  ri  i A e al i  /, <i>ci</i>  o an ct to i	
---	--

<p><i>k</i> 1, — o <i>zi</i> n r, k e n h g e in 9 — <i>ti</i> e <i>on</i> k 2, 1, — et id n A k i al</p>		
---	--	--

<p>k p.</p> <p>i et</p> <p>a g</p> <p>y y i</p> <p>di ne</p> <p>n al</p> <p>k</p> <p>I</p> <p>—</p> <p>l</p> <p>y</p> <p>is of</p> <p>e or</p> <p>i</p> <p>ih o n</p> <p>s )</p> <p>ic</p> <p>l</p>		
---	--	--

re s.  ni s, /-  d  i- s  g  or g  in  ti  n  in d h  /-  ]. ol /- or k  /- 2 0  r  d an l of ks		
---	--	--

<p>en a d l, in ti e on e d e i 0 5, i- i as ar f- i in te , r v. g d lh s., n, n r, - ,</p>	
--	--



--	--	--

s g i o g, 5- p. — es at i P / in si on k N g, g id s g b n in — l es g, e g		
---	--	--

2, -		
n		
n		
ni d b		
iv		
g ne y t		
e g 8, -		
d n, e		
i d c		
it		
r- n		
vi d o		
i e		

--	--	--

v i e s h e  q id h A  l of nt  , ti e on te  it  , e ft ti  r, i- e s, e id n		
---	--	--

<p>f- i</p> <p>th ti</p> <p>e n al</p> <p>k</p> <p>V</p> <p>-</p> <p>s, e</p> <p>n d a</p> <p>n on</p> <p>e</p> <p>y at of</p> <p>at</p> <p>l</p> <p>ks</p> <p>it</p> <p>-</p> <p>n r</p> <p>n, o</p>	
---	--

as  
in  
>  
of  
ta  
  
r  
  
d  
  
a  
  
l,  
  
-  
  
k  
  
n  
  
e  
g  
or  
D.  
].  
ol  
/-  
i  
d  
  
**A.**  
  
*i*  
  
st  
  
of  
e  
  
to  
  
e  
i-

--	--

er os to  o e e n e e of s e  5 6  y,  or  o e n e of e  n  h d  of d, st or l or l n re . 2 3		
--	--	--



present the results of applying the student's t-test to the results showed in Table 1 (classification error); while Tables 6 and 7 represent the same calculation for results of Table 5 (goodness of maps). The results that represent a significant statistical difference (value lower than 0.05) are highlighted.

Table 1.  
Classification  
Error

**Dataset**

		<b>1</b>	<b>2</b>	<b>3</b>	<b>4</b>	<b>5</b>
Cancer	Unsup. SOM	<b>0.0095</b>	<b>0.0099</b>	<b>0.0106</b>	<b>0.0095</b>	<b>0.0099</b>
	Semi-Sup. SOM	0.0198	0.0183	0.0187	0.0176	0.0183
	Ens. Fus. Unsup. SOM	0.0271	0.0267	0.0212	0.0245	0.0245
	Ens. Fus. Semi-Sup. SOM	0.0300	0.0293	0.0249	0.0220	0.0212
	Unsup. GNG	0.0264	<b>0.0223</b>	<b>0.0227</b>	0.0227	0.0242
	Semi-Sup. GNG	<b>0.0256</b>	0.0278	0.0267	0.0271	0.0293
	Ens. Fus. Unsup. GNG	0.0267	0.0282	0.0260	<b>0.0205</b>	<b>0.0216</b>
Ens. Fus. Semi-Sup. GNG	0.0275	0.0289	0.0275	0.0216	0.0220	
Echo-Cardio.	Unsup. SOM	<b>0.37</b>	<b>0.36</b>	<b>0.36</b>	0.38	0.38
	Semi-Sup. SOM	0.38	0.38	0.39	<b>0.37</b>	<b>0.34</b>
	Ens. Fus. Unsup. SOM	0.42	0.41	0.39	0.39	0.36
	Ens. Fus. Semi-Sup. SOM	0.41	0.40	0.39	0.38	0.37
	Unsup. GNG	0.42	0.41	0.40	0.38	0.38
	Semi-Sup. GNG	<b>0.41</b>	0.40	0.39	0.41	0.40
Ens. Fus. Unsup. GNG	0.38	<b>0.36</b>	<b>0.32</b>	<b>0.33</b>	<b>0.28</b>	

	Ens. Fus. Semi-Sup. GNG	0.42	0.39	0.36	0.37	0.31
Glass	Unsup. SOM	<b>0.11</b>	<b>0.11</b>	<b>0.09</b>	<b>0.11</b>	<b>0.11</b>
	Semi-Sup. SOM	0.21	0.21	0.20	0.21	0.27
	Ens. Fus. Unsup. SOM	0.41	0.38	0.33	0.31	0.28
	Ens. Fus. Semi-Sup. SOM	0.38	0.33	0.27	0.24	0.36
	Unsup. GNG	<b>0.20</b>	0.24	<b>0.22</b>	<b>0.23</b>	<b>0.22</b>
	Semi-Sup. GNG	0.22	<b>0.22</b>	0.25	<b>0.23</b>	0.33
	Ens. Fus. Unsup. GNG	0.40	0.34	0.29	0.28	0.23
	Ens. Fus. Semi-Sup. GNG	0.39	0.33	0.28	0.25	0.34

Table 2. Results for the t-student's test performed over the classification results of the different SOM-based models for the Wine dataset.

	Unsup. SOM	Fus. Unsup.SOM	SemiSup. SOM	Fus. SemiSup. SOM
Unsup. SOM	1	0,272	0,587	<b>0,022</b>
Fus. Unsup. SOM		1	0,351	<b>0,033</b>
SemiSup. SOM			1	<b>0,042</b>
Fus. SemiSup. SOM				1

Table 3. Results for the t-student's test performed over the classification results of the different GNG-based models for the Wine dataset.

	Unsup. GNG	Fus. Unsup.GNG	SemiSup. GNG	Fus. SemiSup. GNG
Unsup. GNG	1	0,588	0,347	0,230
Fus. Unsup. GNG		1	0,465	<b>0,028</b>
SemiSup. GNG			1	0,173
Fus. SemiSup. GNG				1

Table 4. Quantization Error

Dataset	Trained Model	number of maps				
		1	2	3	4	5
Cancer	Unsup. SOM	<b>0.0089</b>	0.0089	0.0088	0.0089	0.0089
	Semi-Sup. SOM	<b>0.0089</b>	0.0089	0.0089	0.0088	0.0089
	Ens. Fus. Unsup. SOM	0.0094	0.0087	0.0083	<b>0.0080</b>	<b>0.0079</b>
	Ens. Fus. Semi-Sup. SOM	0.0093	<b>0.0086</b>	<b>0.0082</b>	0.0081	<b>0.0079</b>
	Unsup. GNG	0.0095	0.0086	0.0086	0.0084	0.0079
	Semi-Sup. GNG	0.0102	0.0081	0.0090	0.0078	0.0082
	Ens. Fus. Unsup. GNG	<b>0.0082</b>	0.0084	0.0083	0.0085	0.0084
Echo-Cardio.	Ens. Fus. Semi-Sup. GNG	0.0083	<b>0.0074</b>	<b>0.0069</b>	<b>0.0066</b>	<b>0.0063</b>
	Unsup. SOM	<b>0.0080</b>	0.0091	<b>0.0080</b>	0.0087	0.0093
	Semi-Sup. SOM	0.0084	<b>0.0080</b>	0.0087	<b>0.0080</b>	<b>0.0083</b>
	Ens. Fus. Unsup. SOM	0.0212	0.0139	0.0108	0.0085	0.0089
	Ens. Fus. Semi-Sup. SOM	0.0264	0.0129	0.0111	0.0092	0.0084
	Unsup. GNG	<b>0.0095</b>	0.0086	0.0086	0.0084	0.0079
	Semi-Sup. GNG	0.0102	<b>0.0081</b>	0.0090	0.0078	0.0082
Glass	Ens. Fus. Unsup. GNG	0.0121	0.0102	0.0081	<b>0.0064</b>	0.0060
	Ens. Fus. Semi-Sup. GNG	0.0102	<b>0.0081</b>	<b>0.0077</b>	0.0073	<b>0.0057</b>
	Unsup. SOM	<b>5.25E-06</b>	<b>4.47E-06</b>	4.73E-06	<b>5.13E-06</b>	<b>5.33E-06</b>
	Semi-Sup. SOM	8.99E-06	1.01E-05	9.13E-06	9.52E-06	5.34E-06
	Ens. Fus. Unsup. SOM	2.28E-05	1.23E-05	<b>1.16E-05</b>	9.23E-06	6.67E-06
	Ens. Fus. Semi-Sup. SOM	1.84E-05	1.10E-05	9.34E-06	7.80E-06	6.47E-06
	Unsup. GNG	<b>2.06E-06</b>	<b>1.97E-06</b>	<b>2.15E-06</b>	<b>2.05E-06</b>	<b>2.09E-06</b>
Semi-Sup. GNG	3.15E-06	3.18E-06	3.21E-06	3.07E-06	2.38E-06	
Ens. Fus. Unsup. GNG	1.89E-05	1.50E-05	9.65E-06	8.04E-06	7.85E-06	
Ens. Fus. Semi-Sup. GNG	1.82E-05	1.20E-05	1.02E-05	1.08E-05	6.99E-06	

Table 5. Goodness of Map

Dataset	Trained Model	number of maps				
		1	2	3	4	5
Cancer	Unsup. SOM	0.0318	0.0310	<b>0.0300</b>	0.0306	0.0314
	Semi-Sup. SOM	<b>0.0308</b>	<b>0.0298</b>	0.0313	<b>0.0297</b>	0.0307
	Ens. Fus. Unsup. SOM	0.0993	0.0554	0.0389	0.0327	0.0298
	Ens. Fus. Semi-Sup. SOM	0.0839	0.0511	0.0382	0.0330	<b>0.0287</b>
	Unsup. GNG	0.1377	0.1179	0.1115	0.0927	0.1031
	Semi-Sup. GNG	0.1530	0.1178	0.1165	0.1262	0.1135
	Ens. Fus. Unsup. GNG	0.1123	0.0648	0.0413	0.0321	0.0296
	Ens. Fus. Semi-Sup. GNG	<b>0.1098</b>	<b>0.0529</b>	<b>0.0391</b>	<b>0.0320</b>	<b>0.0274</b>
Echo-Cardio.	Unsup. SOM	0.116	0.138	<b>0.108</b>	0.107	<b>0.097</b>
	Semi-Sup. SOM	<b>0.108</b>	<b>0.111</b>	0.150	0.106	0.159
	Ens. Fus. Unsup. SOM	1.176	0.741	0.230	<b>0.079</b>	0.098
	Ens. Fus. Semi-Sup. SOM	0.941	0.760	0.187	0.151	0.089
	Unsup. GNG	0.368	0.272	0.339	0.194	0.278
	Semi-Sup. GNG	<b>0.340</b>	0.354	0.126	0.302	0.204
	Ens. Fus. Unsup. GNG	0.489	0.178	<b>0.042</b>	<b>0.030</b>	<b>0.024</b>
	Ens. Fus. Semi-Sup. GNG	1.048	<b>0.095</b>	0.100	0.037	<b>0.024</b>
Glass	Unsup. SOM	<b>6.76E-05</b>	6.76E-05	7.95E-05	6.49E-05	7.85E-05
	Semi-Sup. SOM	1.36E-04	1.37E-04	1.21E-04	1.44E-04	7.85E-05
	Ens. Fus. Unsup. SOM	4.06E-04	1.33E-04	7.64E-05	6.10E-05	3.76E-05
	Ens. Fus. Semi-Sup. SOM	1.79E-04	<b>5.86E-05</b>	<b>3.80E-05</b>	<b>3.72E-05</b>	<b>3.41E-05</b>
	Unsup. GNG	<b>1.94E-05</b>	<b>1.79E-05</b>	2.08E-05	1.81E-05	2.38E-05
	Semi-Sup. GNG	2.24E-05	2.07E-05	<b>1.86E-05</b>	<b>1.93E-05</b>	<b>2.03E-05</b>
	Ens. Fus. Unsup. GNG	2.95E-04	1.22E-04	5.42E-05	4.43E-05	3.97E-05
	Ens. Fus. Semi-Sup. GNG	1.96E-04	7.79E-05	5.75E-05	5.82E-05	3.41E-05

Table 6. Results for the t-student's test performed over the goodness of map results of the different SOM-based models for the Wine dataset.

	Unsup. SOM	Fus. Unsup.SOM	SemiSup. SOM	Fus. SemiSup. SOM
Unsup. SOM	1	<b>0,014</b>	0,838	0,374
Fus. Unsup. SOM		1	<b>0,016</b>	<b>0,064</b>
SemiSup. SOM			1	<b>0,376</b>
Fus. SemiSup. SOM				1

Table 7. Results for the t-student's test performed over the goodness of map results of the different GNG-based models for the Wine dataset.

	Unsup. GNG	Fus. Unsup.GNG	SemiSup. GNG	Fus. SemiSup. GNG
Unsup. GNG	1	0,622	0,159	0,203
Fus. Unsup. GNG		1	0,524	<b>0,067</b>
SemiSup. GNG			1	0,178
Fus. SemiSup. GNG				1

## A.2. Experiment 2

The second experiment consists of 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 datasets, as when using a less amount of data the training process becomes more difficult.

Tables 9, 10 and 11 present, respectively, the numerical results for the classification error, the quantization error and the goodness of map for the three additional datasets in the study.

For each dataset and each number of maps used, the lowest error for SOM-based and for GNG-based combinations are highlighted.

Table 8. Classification Error

Dataset	Trained Model	size of dataset				
		5/5	4/5	3/5	2/5	1/5
Cancer	Unsup. SOM	<b>0.010</b>	0.010	0.012	0.014	<b>0</b>
	Semi-Sup. SOM	0.017	<b>0.008</b>	<b>0.008</b>	<b>0.002</b>	0.004
	Ens. Fus. Unsup. SOM	0.019	0.009	0.015	0.008	0.024
	Ens. Fus. Semi-Sup. SOM	0.014	0.020	0.016	0.014	0.036
	Unsup. GNG	0.026	0.021	0.020	0.016	0.014
	Semi-Sup. GNG	0.028	0.018	0.017	0.012	<b>0.012</b>
	Ens. Fus. Semi-Sup. GNG	<b>0.013</b>	<b>0.014</b>	<b>0.009</b>	<b>0.004</b>	0.024
Echo-Cardio.	Unsup. SOM	0.38	<b>0.33</b>	<b>0.37</b>	0.36	<b>0.26</b>
	Semi-Sup. SOM	<b>0.34</b>	0.34	<b>0.37</b>	<b>0.34</b>	<b>0.26</b>
	Ens. Fus. Unsup. SOM	0.36	0.38	0.43	0.40	0.29
	Ens. Fus. Semi-Sup. SOM	0.37	0.37	0.43	0.36	0.38
	Unsup. GNG	0.38	0.39	0.40	0.36	0.23
	Semi-Sup. GNG	0.40	0.37	0.40	0.36	0.38
	Ens. Fus. Semi-Sup. GNG	<b>0.28</b>	<b>0.27</b>	<b>0.30</b>	<b>0.25</b>	<b>0.22</b>
Glass	Unsup. SOM	0.110	<b>0.052</b>	<b>0.066</b>	<b>0.039</b>	<b>0.022</b>
	Semi-Sup. SOM	<b>0.27</b>	0.28	0.40	0.23	0.037
	Ens. Fus. Unsup. SOM	0.28	0.27	0.31	0.27	0.40
	Ens. Fus. Semi-Sup. SOM	0.36	0.36	0.51	0.23	0.38
	Unsup. GNG	<b>0.22</b>	<b>0.19</b>	<b>0.19</b>	<b>0.13</b>	0.09
	Semi-Sup. GNG	0.33	0.32	0.45	0.10	<b>0.03</b>
	Ens. Fus. Semi-Sup. GNG	0.23	0.22	0.27	0.23	0.38
	Ens. Fus. Unsup. GNG	0.34	0.35	0.45	0.22	0.33

Table 9. Quantization Error

Dataset	Trained Model	size of dataset				
		5/5	4/5	3/5	2/5	1/5
Cancer	Unsup. SOM	0.0088	0.0086	0.0083	0.0103	<b>0.0077</b>
	Semi-Sup. SOM	0.0088	0.0086	0.0083	0.0086	<b>0.0077</b>
	Ens. Fus. Unsup. SOM	<b>0.0079</b>	<b>0.0060</b>	0.0075	0.0092	0.0083
	Ens. Fus. Semi-Sup. SOM	<b>0.0079</b>	0.00769	<b>0.0074</b>	<b>0.0078</b>	<b>0.0077</b>
	Unsup. GNG	0.0080	0.0077	0.0075	0.0065	<b>0.0057</b>
	Semi-Sup. GNG	0.0080	0.0077	0.0075	0.0073	<b>0.0057</b>
	Ens. Fus. Unsup. GNG	0.0064	0.0077	0.0058	0.0061	0.0079
	Ens. Fus. Semi-Sup. GNG	<b>0.0063</b>	<b>0.0060</b>	<b>0.0057</b>	<b>0.0058</b>	0.0073
Echo-Cardio.	Unsup. SOM	0.00927	<b>0.0082</b>	0.0081	<b>0.0074</b>	<b>0.0049</b>
	Semi-Sup. SOM	<b>0.0082</b>	<b>0.0082</b>	<b>0.0075</b>	0.0080	0.082
	Ens. Fus. Unsup. SOM	0.0085	0.0105	0.0081	0.0103	0.0049
	Ens. Fus. Semi-Sup. SOM	0.0083	0.0150	0.0119	0.0182	0.0105
	Unsup. GNG	0.0079	0.0076	0.0071	0.0068	0.0037
	Semi-Sup. GNG	0.00817	<b>0.00706</b>	<b>0.0067</b>	<b>0.0064</b>	<b>0.0070</b>
	Ens. Fus. Unsup. GNG	0.0059	0.0064	<b>0.0062</b>	0.0137	0.0199
	Ens. Fus. Semi-Sup. GNG	<b>0.0057</b>	0.0078	0.0075	0.0165	0.0185
Glass	Unsup. SOM	<b>5.32E-06</b>	<b>3.67E-06</b>	3.03E-06	4.00E-06	<b>1.78E-06</b>
	Semi-Sup. SOM	5.34E-06	4.02E-06	5.89E-06	3.32E-06	6.42E-06
	Ens. Fus. Unsup. SOM	6.66E-06	8.64E-06	<b>1.06E-05</b>	<b>1.70E-05</b>	2.16E-05
	Ens. Fus. Semi-Sup. SOM	6.46E-06	7.99E-06	1.45E-05	<b>1.04E-05</b>	1.94E-05
	Unsup. GNG	<b>2.08E-06</b>	<b>1.85E-06</b>	1.69E-06	1.59E-06	9.92E-07
	Semi-Sup. GNG	2.37E-06	2.45E-06	3.89E-06	1.89E-06	<b>5.84E-07</b>
	Ens. Fus. Unsup. GNG	7.85E-06	8.92E-06	1.37E-05	1.29E-05	1.79E-05
	Ens. Fus. Semi-Sup. GNG	6.98E-06	6.53E-06	<b>1.27E-05</b>	1.26E-05	2.64E-05

Table 10. Goodness of Maps

Dataset	Trained Model	size of dataset				
		5/5	4/5	3/5	2/5	1/5
Cancer	Unsup. SOM	0.031	0.026	0.022	0.045	<b>0.012</b>
	Semi-Sup. SOM	0.030	<b>0.024</b>	<b>0.021</b>	<b>0.018</b>	<b>0.012</b>
	Ens. Fus. Unsup. SOM	0.029	0.028	0.027	0.028	0.037
	Ens. Fus. Semi-Sup. SOM	<b>0.028</b>	0.027	0.026	0.028	0.033
	Unsup. GNG	0.103	0.092	0.065	0.041	<b>0.029</b>
	Semi-Sup. GNG	0.113	0.102	0.085	0.061	<b>0.029</b>
	Ens. Fus. Unsup. GNG	0.029	0.025	0.023	0.031	0.047
	Ens. Fus. Semi-Sup. GNG	<b>0.027</b>	<b>0.024</b>	<b>0.021</b>	<b>0.021</b>	0.036
Echo-Cardio.	Unsup. SOM	0.097	<b>0.081</b>	<b>0.072</b>	<b>0.068</b>	<b>0.027</b>
	Semi-Sup. SOM	0.159	0.084	0.091	0.071	0.069
	Ens. Fus. Unsup. SOM	0.098	0.236	0.077	0.107	<b>0.067</b>
	Ens. Fus. Semi-Sup. SOM	<b>0.089</b>	0.096	0.112	0.149	0.0147
	Unsup. GNG	0.278	0.295	0.123	0.076	<b>0.030</b>
	Semi-Sup. GNG	0.204	0.165	0.083	0.101	0.095
	Ens. Fus. Unsup. GNG	<b>0.024</b>	<b>0.028</b>	<b>0.024</b>	<b>0.054</b>	0.106
	Ens. Fus. Semi-Sup. GNG	<b>0.024</b>	<b>0.028</b>	0.208	0.079	0.082
Glass	Unsup. SOM	7.85E-05	5.96E-05	7.56E-05	7.84E-05	<b>3.97E-05</b>
	Semi-Sup. SOM	7.19E-05	8.15E-05	9.45E-05	8.56E-05	6.39E-05
	Ens. Fus. Unsup. SOM	3.76E-05	4.03E-05	<b>6.37E-05</b>	9.01E-05	1.74E-04
	Ens. Fus. Semi-Sup. SOM	<b>3.41E-05</b>	<b>3.36E-05</b>	7.73E-05	<b>5.30E-05</b>	1.11E-04
	Unsup. GNG	2.38E-05	<b>1.54E-05</b>	<b>1.43E-05</b>	1.46E-05	<b>1.08E-05</b>
	Semi-Sup. GNG	<b>2.03E-05</b>	1.89E-05	2.20E-05	<b>1.32E-05</b>	6.36E-06
	Ens. Fus. Unsup. GNG	3.97E-05	4.33E-05	7.17E-05	4.85E-05	1.10E-04
	Ens. Fus. Semi-Sup. GNG	3.41E-05	3.42E-05	6.28E-05	6.08E-05	1.24E-04

## Appendix B. Training parameters

In all experiments the parameters used for the Voronoi Polygon Similarity Fusion are those appearing in Table 11. The parameters for the base classifiers in

each experiment are detailed in Table 12. They have been left unchanged for all experiments presented in the current research.

Table 11. Parameters used for the Ensemble Fusion

UsageThreshold	2/number of Samples
Fusion Threshold	0.5
Connection Threshold	0.8



Table 12. Parameters used for the base learners with the different datasets

Dataset	Base Learn.	Parameters	Dataset	Base Learn.	Parameters
Iris	SOM	Size = 15x10 Times = 1500 Learn. Rate = 0.1 Gaussian Neighb.: c1 = 1, Sigma = 10	Echo- Cardiogram	SOM	Size = 15x10 Times = 2000 Learn. Rate = 0.05 Diff. Gaussian Neighb.: c1 = 2, Sigma1 = 10, c2 = 1, Sigma2 = 14
	GNG	Times = 500 Maximum Edge Age = 3 Lambda = 10 Alpha = 0.3 Beta = 0.1 Winner Learn. Rate = 0.1 Neighb. Learn. Rate = 0.005		GNG	Times = 500 Maximum Edge Age = 2 Lambda = 10 Alpha = 0.01 Beta = 0.0005 Winner Learn. Rate = 0.05 Neighb. Learn. Rate = 0.005
Wine	SOM	Size = 15x10 Times = 1800 Learn. Rate = 0.05 Diff. Gaussian Neighb.: c1 = 2, Sigma1 = 10, c2 = 1, Sigma2 = 12	Glass	SOM	Size = 25x20 Times = 2000 Learn. Rate = 0.1 Diff. Gaussian Neighb.: c1 = 2, Sigma1 = 12, c2 = 1, Sigma2 = 14
	GNG	Times = 1500 Maximum Edge Age = 2 Lambda = 10 Alpha = 0.2 Beta = 0.1 Winner Learn. Rate = 0.1 Neighb. Learn. Rate = 0.005		GNG	Times = 2000 Maximum Edge Age = 2 Lambda = 10 Alpha = 0.2 Beta = 0.075 Winner Learn. Rate = 0.2 Neighb. Learn. Rate = 0.01
Cancer	SOM	Size = 25x20 Times = 5000 Learn. Rate = 0.1 Diff. Gaussian Neighb.: c1 = 1, Sigma1 = 10			
	GNG	Times = 1000 Maximum Edge Age = 2 Lambda = 10 Alpha = 0.5 Beta = 0.2 Winner Learn. Rate = 0.1 Neighb. Learn. Rate = 0.005			