EnvSOM: A SOM Algorithm Conditioned on the Environment for Clustering and Visualization

Serafín Alonso[1], Mika Sulkava[2], Miguel Angel Prada[2], Manuel Domínguez[1], and Jaakko Hollmén[2]

[1] Grupo de Investigación SUPPRESS, Universidad de León, León, Spain
saloc@unileon.es, manuel.dominguez@unileon.es
[2] Department of Information and Computer Science, Aalto University School of Science, Espoo, Finland
mika.sulkava@tkk.fi, miguel.prada@tkk.fi, Jaakko.Hollmen@hut.fi

Abstract. In this paper, we present a new approach suitable for analysis of large data sets, conditioned on the environment. Mainly, the envSOM algorithm consists of two consecutive trainings of the self-organizing map. In the first phase, a SOM is trained using every available variable, but only those which characterize the environment are used to compute the winner unit. Therefore, this phase produces an accurate model of the environment. In the second phase, a new SOM is initialized appropriately with information from the codebooks of the first SOM. The new SOM uses all the variables for winner selection. However, in this case the environmental variables are kept fixed and only the remaining ones are involved in the update process. A model of the whole data set influenced by the environmental conditions is obtained in this second phase. The result of this algorithm represents a probability function of a data set, given the environment information. Therefore, it could be very useful in the analysis of processes which have close dependencies on environmental conditions.

Keywords: Self-organizing maps, variants of SOM, environmental conditions, envSOM, data mining, pattern recognition.

1 Introduction

Many variants of SOM appeared in the literature [1]. The aims of these approaches comprise improvements in clustering, visualization, accuracy of the model, computation time, etc. For instance, it is possible to define different neighborhood functions, change the winner searching process, and introduce some a priori information about classes or states. An overview of the main ideas which can be used to modify the standard SOM is presented in [2].

These variants have brought great advantages for data analysis, but, so far, none of them has been focused on the data analysis conditioned on the environment. It is well known that environmental conditions influence strongly most of the real processes and systems. Furthermore, it is generally desirable to compare

J. Laaksonen and T. Honkela (Eds.): WSOM 2011, LNCS 6731, pp. 61–70, 2011.

data from different processes whose environmental conditions are the same. For these reasons, a new algorithm, the envSOM, is proposed in this paper. It still captures the behavior of the processes, but takes into account the model of the environment.

This paper is structured as follows: In Section 2, several approaches related to the envSOM are reviewed briefly. In Section 3, the envSOM algorithm and its two phases are explained in detail. Two examples used to test the algorithm are described in Section 4. Also, the results obtained using the envSOM are shown there. Finally, the conclusions are drawn in Section 5.

2 Similar Approaches

Several approaches, already presented in the literature, are described below. Although they have some similarities with the proposed algorithm, they also have essential differences.

In the *Supervised SOM* [3], the main idea is to modify the traditional unsupervised SOM into a supervised algorithm by adding information about class-identity in the learning process. For that purpose, the input vectors, $x = [x_s, x_u]$, consist of two different parts. The first one, x_s, corresponds to the input data and the second one, x_u, is related to the class of the sample [4]. In order to visualize the map, the second part is pruned out. Classification is enhanced using this method since the second part is the same for input vectors of the same class. It could be necessary to weight the values of the second part to achieve a better accuracy in the classification. The proposed envSOM algorithm does not depend on a class-identity variable, but the input vectors comprise two rather different parts, the environmental variables and the others.

The *Tree-Structured SOM* (TS-SOM) consists of several traditional SOMs organized hierarchically in several layers, i.e., a pyramid-like structure is obtained where the lower SOMs are larger [5,6]. Firstly, training will take place at the higher levels. Codebooks from these SOMs are kept fixed and then, the training continues at the subsequent layers according to the hierarchy. The differences appear in both search and update steps. The winner searching process is performed on the units at the same layer and the neighbors on the higher level according to the hierarchy. In the update step, only the units at the same layer are updated, keeping fixed the units at the higher level. Therefore, this variant is computationally quite inexpensive whereas the envSOM comprises two consecutive SOM trainings.

The *PicSOM* algorithm is based on several TS-SOMs [7]. It was proposed for retrieving images similar to a given reference image from a database. A separate TS-SOM is used for each kind of feature vectors extracted from the images (color, texture and shape). The responses from individual TS-SOMs are combined automatically according to user's preferences. The PicSOM approach provides a robust method for using a set of image maps in parallel. The envSOM algorithm also uses a set of special variables (characterizing the environment) to cluster data.

In the *Layering SOM*, a SOM is trained for each individual layer to achieve better results in the field of exploratory analysis. In this sense, a growing hierarchical SOM has been presented in [8]. That work explains a dynamic model which adapts its architecture in the training process and uses more units where more input data are projected. The major benefits of this approach are the reduction of training time due to the concept of layers, the possibility to discover a hierarchical structure of the data, the improvement of cluster visualization by displaying small maps at each layer and the preservation of topological similarities between neighbors. This algorithm allows us to visualize data in detail, but the models obtained are not conditioned on the environment.

The self-organizing map has also been used for time series processing in the form of the *Temporal SOM*. In order to exploit the temporal information, SOM needs to be enabled with a short-term memory, which can be implemented, e.g., through external tapped delay lines or different types of recurrence. Several of these extensions are reviewed in [9,10]. In our approach, no short-term memory is explicitly implemented, but it is usually advisable to introduce time information in the model to analyze the temporal evolution, together with the environment.

3 The envSOM Algorithm

The purpose of this work is to develop an algorithm suitable for extracting and analyzing information from large data sets, but considering the environmental information such as weather variables, atmospheric deposition, etc. The envSOM approach consists of two consecutive phases based on the traditional SOM [2]. Some slight variations have been introduced in each phase. The winner searching process in the first phase and the update process in the second one have been modified appropriately in order to achieve the desired result. In our experiments the learning rate decreases in time and the neighborhood function is implemented as Gaussian in both phases. However, other functions could be used as well.

The proposed envSOM algorithm has the advantageous features of the traditional SOM. Likewise, it reaches spatially-ordered and topology-preserving maps. It also provides a good approximation to the input space, similar to vector quantization, and divides the space in a finite collection of Voronoi regions. The main innovation of this algorithm is that it reflects the probability density function of data set, given the environmental conditions. Therefore, it can be useful from the point of view of environmental pattern recognition and data comparison, conditioned on these patterns. On the contrary, it should be noted that it will be more expensive computationally compared to the traditional SOM, since two learning phases are needed. Furthermore, it requires knowledge of the environmental variables which influence the behavior of the process, characterized by the remaining variables. The envSOM approach will be explained in detail below.

3.1 The First Phase

In the first phase of the envSOM algorithm, a traditional SOM is trained using all variables. The initialization can be either linear along the greatest eigenvectors or

random, depending on the user's preference. It is necessary to know in advance which are the environmental variables, since only these variables will be used for computing the winner neurons. For this reason, the other variables must be masked in the winner searching process. Similarly to the traditional SOM, the winner c is selected using equation 1.

$$c(t) = \arg\min_i \|\mathbf{x}(t) - \mathbf{m}_i(t)\|_\omega \ , i = 1, 2, \ldots, N \tag{1}$$

where $\mathbf{x}$ represents the current input and $\mathbf{m}$ denotes the codebook vectors. N and t are, respectively, the number of the map units and the time. The difference is that a binary mask is always used to indicate which variables are used for computing the winner. As usual, if the Euclidean norm, $\|^\circ\|$, is chosen, the winner will be computed using equation 2, where ω is the binary mask and k is a component or variable.

$$\|\mathbf{x}(t) - \mathbf{m}_i(t)\|_\omega^2 = \omega\|\mathbf{x}(t) - \mathbf{m}_i(t)\|^2 = \sum_k \omega_k \left[x_k(t) - m_{ik}(t)\right]^2 \tag{2}$$

The mask, ω, is a k-dimensional vector whose values ω_k are 1 or 0, depending on if the component corresponds to an environmental variable or not. The update rule has not been modified and therefore, it is similar to the traditional SOM.

The result obtained from this phase will be a map where only the components related to the environment are organized. The remaining components do not affect the organization. The aim of this phase is to achieve a model which represents the environment in the best possible way. Moreover, the values of the remaining components will be used for initialization in the second phase. It should be remarked that although this initialization seems completely random, it has proven to be good and the values lie in the range of the variables.

3.2 The Second Phase

In the second phase of the envSOM algorithm, a new traditional SOM is trained using all variables. It will be initialized using the codebooks from the first phase SOM. Thanks to this appropriate initialization, a fast convergence of the algorithm is reached and an accurate model which defines the environment will be used in the second phase. It should be noted that environmental components have been already organized in the first phase of the envSOM. Therefore, values from the codebooks of first SOM are a good starting point for the second phase.

In this case, every component will take part equally in the winner computation and no mask will be applied. Unlike the first phase, the update process is now slightly modified. As environmental variables are already well organized, it is only required that the remaining variables are updated properly. For this reason, a new mask is introduced in the update rule and equation 3 will be used in this case. The mask, $\mathbf{\Omega}$, is a k-dimensional vector which takes binary values Ω_k, i.e., 0 if it corresponds to an environmental variable and 1 otherwise. k is the number of components or variables.

$$\mathbf{m}_i(t+1) = \mathbf{m}_i(t) + \alpha(t) h_{ci}(t) \mathbf{\Omega}[\mathbf{x}(t) - \mathbf{m}_i(t)] \tag{3}$$

At the end of this phase, all variables will be organized properly. The learning rate, $\alpha(t)$, and the neighborhood function, $h_{ci}(t)$, are not modified so that a value decreasing in time and a Gaussian function could be used, respectively, like in the traditional SOM. The purpose of this phase is to reach a good model of the whole data set, given environmental information.

4 Experiments and Results

Two kinds of experiments have been planned in order to test the envSOM algorithm. First, an artificial data set based on binary patterns is created. It allows us to check the clustering property of the algorithm. Then, a simulated data set characterizing climate and carbon flux in several ecosystems is studied. It allows us to check the usefulness of the algorithm with more realistic data and compare the behavior of carbon in different ecosystems, given environmental conditions.

Matlab software has been used to make the experiments and the SOM Toolbox [11] has been modified to implement the necessary changes, such as a new mask in the update process.

4.1 A Toy Example

An artificial data set with structured data has been used to test the envSOM algorithm. The data set consists of 16000 samples and 4 variables (X1, X2, X3, X4). It contains all binary patterns from (0, 0, 0, 0) to (1, 1, 1, 1), i.e., the numbers from 0 to 15 in binary system. A low level of noise (10%) has been added to the variables. Each binary pattern is equally represented by a set of 1000 samples. There are 16 different patterns, so the envSOM algorithm should find 16 clusters in this data set. The choice of this data set is justified by the simple structure of the data, which facilitates the visualization and understanding of the results from the algorithm.

First, a traditional SOM was trained using this input data set. The number of epochs in the training should be high enough in order to guarantee a complete organization. A number over 500 epochs was chosen. The dimensions of SOM were 16×20 (320 units). A Gaussian function was selected as the neighborhood function and a value decreasing exponentially in time as the learning rate. The SOM should be able to divide the data into 16 clusters and allows us to visualize them, for instance, by means of the U-matrix representation. The results from the traditional SOM can be seen in Figure 1. After the training, the U-matrix yields a clear visualization of the binary patterns. Note that each component has been organized in a random way, as it is shown by the component planes. If a new traditional SOM is trained using another data set Y, also based on binary patterns, i.e., (Y1, Y2, Y3, Y4), the organization of the four components will probably be completely different. Thus, it will be very difficult to make a good comparison between the results from both data sets, X and Y.

When there are environmental conditions in the data set, it can be desirable that these components define the organization of the map. In this case, it is supposed that X1 and X2 are the environmental variables and X3 and X4 are

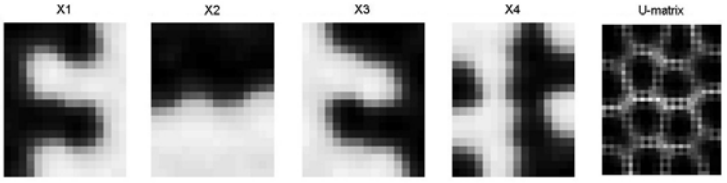

Fig. 1. Component planes and U-matrix of traditional SOM for binary patterns. Black color corresponds to values of 0 and white color to 1.

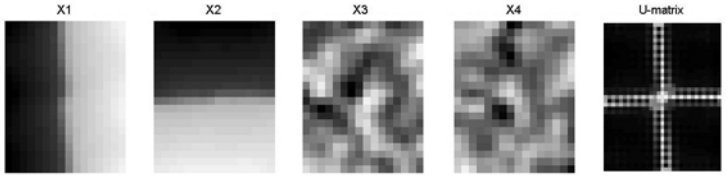

Fig. 2. Component planes and U-matrix of envSOM algorithm after the first phase of learning

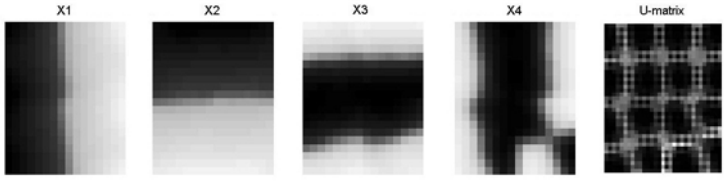

Fig. 3. Component planes and U-matrix of envSOM algorithm after the second phase of learning

features of the data set to be analyzed and compared. The envSOM algorithm consists of two consecutive SOMs as mentioned above. The parameters of both SOMs are the same as in the traditional SOM (500 epochs, 320 units, Gaussian neighborhood function and learning rate decreasing exponentially).

In the first phase, only X1 and X2 variables are used to compute the winner neurons and all variables are updated. The results of the first phase can be seen in Figure 2. As expected, the organization is only performed on variables X1 and X2 since X3 and X4 do not take part in the winner computation. Therefore, the U-matrix only represents four patterns corresponding to possible combinations of variables X1 and X2.

In the second phase, all four variables are used in the winner computation, but X1 and X2 are kept fixed whereas X3 and X4 are updated. At the end of this phase, the data set is organized as depicted in Figure 3. In this case, the 16 patterns can be clearly distinguished in the U-matrix in a similar way to the traditional SOM. Moreover, the organization of the map conditioned on X1 and X2, i.e., the environmental variables, is achieved. It can be said that the envSOM algorithm represents the probability function of data, given the

environmental variables. A comparison of quality of the traditional SOM and the proposed algorithm has been done. The mean quantization error is 0.1364 for the traditional SOM and 0.1717 for the envSOM.

If an envSOM is trained using another binary data set, e.g., Y (Y1, Y2, Y3, Y4), components Y3 and Y4 will be conditioned on the first ones, Y1 and Y2, as expected. The organization of the maps from data sets X and Y can be different and therefore the comparison is difficult. However, the result after the first phase of the envSOM with the data set X can be used to organize Y3 and Y4 in the same way that X3 and X4, respectively. Now, it will be very easy to compare the results from both data sets, X and Y. Furthermore, when components Y1 and Y2 are the same as X1 and X2 because they represent the common environmental conditions, the first phase of the envSOM can be trained jointly with the variables X1, X2, X3, X4, Y3, Y4. The second phase of the envSOM can be carried out with an individual SOM for each data set or one SOM containing all variables from both data sets. The first approach can be applied in any case but, whenever the number of variables and data sets is low enough, the second approach will provide similar results. In those cases, the second choice might be preferred, since it requires fewer computations.

4.2 O-CN Example

A more realistic scenario for presenting the performance of the envSOM approach was performed by analyzing data containing environmental characteristics and simulated gross primary production (GPP, the amount of carbon sequestrated in photosynthesis) of different ecosystems in Europe. The SOM has been previously used for analysis of carbon exchange of ecosystems in, e.g., [12,13]. The GPP estimates used in this study has been generated by the O-CN model [14,15]. The model is developed from the land surface scheme ORCHIDEE [16], and has been extended through representation of key nitrogen cycle processes. O-CN simulates the terrestrial energy, water, carbon, and nitrogen budgets for discrete tiles (i.e. fractions of the grid cell) occupied by up to 12 plant functional types (PFTs) from diurnal to decadal timescales. The model can be run on any regular grid, and is applied here at a spatial resolution of 0.5×0.5. Values of the model input variables: air temperature, precipitation, shortwave downward flux, longwave downward flux, specific humidity, and N deposition and simulated GPP from 1996 to 2005 were used in this example. These values were analyzed for four PFTs: temperate needle-leaved evergreen forests (TeNE), temperate broadleaved seasonal forests (TeBS), temperate grasslands (TeH), and temperate croplands (TeH crop).

The envSOM algorithm was compared with the traditional SOM in this example. First, four traditional SOMs were trained for four PFTs using environmental data and GPP estimates. As example, the component planes and U-matrices of SOMs of two PFTs, temperate broadleaved seasonal forests and temperate grasslands, are shown in Figures 4 and 5. The organization of the two maps characterizing the two PFTs is very different from each other, so it is very laborious to compare them. If one tries to compare the magnitudes of GPP in

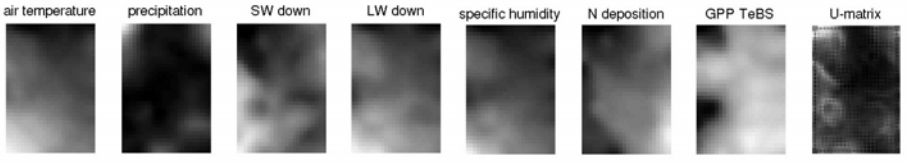

Fig. 4. Component planes and U-matrix of traditional SOM for temperate broadleaved seasonal forests

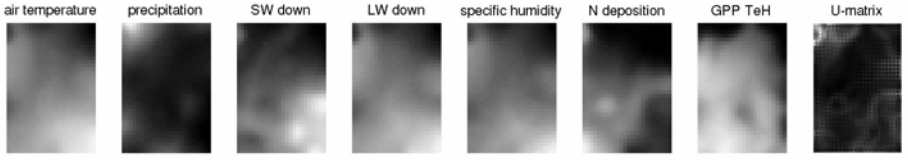

Fig. 5. Component planes and U-matrix of traditional SOM for temperate grasslands

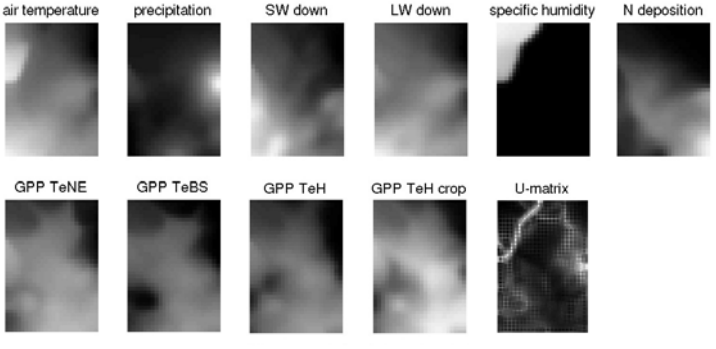

Fig. 6. Component planes and U-matrix of envSOM algorithm for four PFTs

different PFTs connected to a certain combination of environmental variables, spotting the corresponding locations on the maps is not straightforward. The organization of the SOMs of the two PFTs not shown was also different from the other PFTs. The four PFTs were used to train an envSOM with six environmental variables and four GPPs. In the first phase, the environmental variables (air temperature, precipitation, shortwave downward flux, longwave downward flux, specific humidity, and N deposition) were used for training. In the second phase, the variables affected by the environment, i.e., four GPPs were trained. Figure 6 shows the obtained component planes and the U-matrix.

When using envSOM for comparing the PFTs as shown in Figure 6, the commonalities and differences in the connections between environmental variables and GPP can be spotted with ease among the PFTs. The qualitative behavior of the PFTs seems to be rather similar, i.e., relatively high and low GPP values are

usually found in the same regions of the map and are thus, connected with similar environmental conditions. This similarity between the PFTs was expected. However, the absolute values of GPP are different. There are also some differences visible between the PFTs. E.g., the map units with the highest precipitation have very low GPP values for all PFTs except the temperate croplands. In addition, the area in the lower left part of the map associated with relatively high temperature, shortwave and longwave downward fluxes, low precipitation and high GPP in temperate needle-leaved evergreen forests, temperate grasslands, and temperate croplands contains very low values of GPP in temperate broadleaved seasonal forests. The reason for these differences may be different spatial distribution of the PFTs and that some spatially correlated confounding factors have an effect on GPP. More detailed investigation of the reasons behind the differences might be a topic of a future study.

5 Conclusions

In this paper the envSOM algorithm, which is conditioned on the environment was introduced. It consists of two phases based on the traditional SOM. The envSOM has similar features to the traditional SOM in clustering and visualization, although it adds an innovation very useful for finding patterns conditioned on the environment in large data sets. The main innovation of the envSOM is that it represents the data set given the environmental conditions. Therefore, the algorithm is suitable for data analysis of real processes strongly influenced by the environment. On the contrary, it is slightly more expensive computationally, since two consecutive SOMs are trained. The proposed algorithm has been satisfactorily tested using a binary data sets and environmental data and simulated carbon flux estimates of four plant functional types. This algorithm yields similar results in a round of different trainings with the same data set. The environmental variables are always organized in a similar way and the others are conditioned on the first ones. Incremental training is possible using the envSOM, i.e., new features or even data samples could be added to the training later, while keeping the environmental variables the same.

Acknowledgments. We thank Sönke Zaehle for providing us with the O-CN data for this study and insightful comments regarding the analysis of the data.

References

1. Kangas, J., Kohonen, T., Laaksonen, J.: Variants of self-organizing maps. IEEE Transactions on Neural Networks 1, 93–99 (1990)
2. Kohonen, T.: Self-Organizing Maps. Springer, Heidelberg (1995)
3. Hagenbuchner, M., Tsoi, A.C.: A supervised training algorithm for self-organizing maps for structures. Pattern Recognition Letters 26, 1874–1884 (2005)
4. Melssen, W., Wehrens, R., Buydens, L.: Supervised Kohonen networks for classification problems. Chemometrics and Intelligent Laboratory Systems 83, 99–113 (2006)

5. Koikkalainen, P., Oja, E.: Self-organizing hierarchical feature maps. In: International Joint Conference on Neural Networks, vol. 2, pp. 279–284. IEEE, INNS (1990)
6. Koikkalainen, P.: Progress with the tree-structured self-organizing map. In: Cohn, A.G. (ed.) 11th European Conference on Artificial Intelligence, ECCAI (1994)
7. Laaksonen, J., Koskela, M., Laakso, S., Oja, E.: PicSOM - content-based image retrieval with self-organizing maps. Pattern Recognition Letters 21, 1199–1207 (2000)
8. Rauber, A., Merkl, D., Dittenbach, M.: The growing hierarchical self-organizing map: exploratory analysis of high-dimensional data. IEEE Transactions on Neural Networks 13(6), 1331–1341 (2002)
9. Hammer, B., Micheli, A., Sperduti, A., Strickert, M.: Recursive self-organizing network models. Neural Networks 17, 1061–1085 (2004)
10. Guimarães, G., Sousa-Lobo, V., Moura-Pires, F.: A taxonomy of self-organizing maps for temporal sequence processing. Intelligent Data Analysis (4), 269–290 (2003)
11. Vesanto, J., Himberg, J., Alhoniemi, E., Parhankangas, J.: SOM toolbox for Matlab 5 (2000)
12. Abramowitz, G., Leuning, R., Clark, M., Pitman, A.: Evaluating the performance of land surface models. Journal of Climate 21(21), 5468–5481 (2008)
13. Luyssaert, S., Janssens, I.A., Sulkava, M., Papale, D., Dolman, A.J., Reichstein, M., Hollmén, J., Martin, J.G., Suni, T., Vesala, T., Loustau, D., Law, B.E., Moors, E.J.: Photosynthesis drives anomalies in net carbon-exchange of pine forests at different latitudes. Global Change Biology 13(10), 2110–2127 (2007)
14. Zaehle, S., Friend, A.D.: Carbon and nitrogen cycle dynamics in the o-cn land surface model: 1. model description, site-scale evaluation, and sensitivity to parameter estimates. Global Biogeochemical Cycles 24 (February 2010)
15. Zaehle, S., Friend, A.D., Friedlingstein, P., Dentener, F., Peylin, P., Schulz, M.: Carbon and nitrogen cycle dynamics in the o-cn land surface model: 2. role of the nitrogen cycle in the historical terrestrial carbon balance. Global Biogeochemical Cycles 24 (February 2010)
16. Krinner, G., Viovy, N., de Noblet-Ducoudre, N., Ogee, J., Polcher, J., Friedlingstein, P., Ciais, P., Sitch, S., Prentice, I.C.: A dynamic global vegetation model for studies of the coupled atmosphere-biosphere system. Global Biogeochemical Cycles 19 (February 2005)