Engineering Applications of Artificial Intelligence 59 (2017) 218–235

Contents lists available at ScienceDirect

Engineering Applications of Artificial Intelligence

journal homepage: www.elsevier.com/locate/engappai

Beta Scale Invariant Map

Héctor Quintián*, Emilio Corchado

Department of Computer Science and Automation, University of Salamanca, Plaza de la Merced s/n, Salamanca 37007, Spain

ARTICLE INFO

Keywords:
Topology preserving maps
Quality measures
SIM
MLHL-SIM
ViSOM
GNG
Beta distribution
Imbalanced datasets

ABSTRACT

In this study we present a novel version of the Scale Invariant Map (SIM) called Beta-SIM, developed to facilitate the clustering and visualization of the internal structure of complex datasets effectively and efficiently. It is based on the application of a family of learning rules derived from the Probability Density Function (PDF) of the residual based on the beta distribution, when applied to the Scale Invariant Map. The Beta-SIM behavior is thoroughly analyzed and successfully demonstrated over 2 artificial and 16 real datasets, comparing its results, in terms of three performance quality measures with other well-known topology preserving models such as Self Organizing Maps (SOM), Scale Invariant Map (SIM), Maximum Likelihood Hebbian Learning-SIM (MLHL-SIM), Visualization Induced SOM (ViSOM), and Growing Neural Gas (GNG). Promising results were found for Beta-SIM, particularly when dealing with highly complex datasets.

1. Introduction

Among the great variety of tools for multidimensional data visualization, several of the most widely used are those belonging to the family of the topology preserving maps (Chen et al., 2013; Fuertes et al., 2010; Kohonen, 1998; Mohebi and Bagirov, 2016; Wu et al., 2011). Probably the best known among these algorithms is the Self-Organizing Map (SOM) (Chen et al., 2013; Kohonen, 1998, 2013; Haimoudi et al., 2016). It is based on a type of unsupervised learning called competitive learning; an adaptive process in which the units in a neural network gradually become sensitive to different input categories or sets of samples in a specific domain of the input space. The main feature of the SOM algorithm is its topology preservation. When not only the winning unit, but also its neighbors on the lattice are allowed to learn, neighboring units gradually specialize to represent similar inputs, and the representations become ordered on the map lattice.

Several extensions of SOM can be found in the literature such as the Generative Topographic Mapping (GTM) (Bishop et al., 1998; Ghassany and Bennani, 2015), which was developed by Bishop et al. as a probabilistic version of the SOM, in order to overcome some of its limitations, particularly the lack of an objective function. An important application of the GTM is to allow a simpler visualization of high-dimensional data.

Another extension of SOM is the Topographic Product of Experts (ToPoE), and the Harmonic Topographic Map (HaToM) (Fyfe, 2005; Jeong et al., 2015), where the topology preserving map is created from a product of experts.

The use of ensembles with SOM (Akhand and Murase, 2012; Cho, 2000; Dietterich, 2000; Wang and Gupta, 2015) has also been studied to increase the stability and performance of a specific algorithm. One of the most recent developments of ensembles, in the field of topology preserving maps, is the Weighted Voting Superposition (WeVoS) (Baruque and Corchado, 2014). The principal idea is to obtain the final units of the map by a weighted voting among the units in the same position in different maps, according to a quality measure.

The Visualization Induced SOM (ViSOM) (Corchado and Baruque, 2012; Huang and Yin, 2009), is a SOM extension proposed for the direct preservation of the local distance information on the map, along with the topology. The ViSOM constrains the lateral contraction forces between units and hence regularizes the inter-unit distances, so that distances between units in the data space are in proportion to those in the input space. The ViSOM not only takes into account the distance between a unit's weights from one iteration to the next, but also the distance between that unit and the Best Matching Unit within the whole map (BMU). This allows the ViSOM to preserve topology by maintaining distance between neighbors of the winner unit.

Two other interesting topology preserving models are the Scale Invariant Map (SIM) (Baruque and Corchado, 2014, 2009) and the Maximum Likelihood Scale Invariant Map (MLHL-SIM) (Baruque and Corchado, 2011; Corchado and Fyfe, 2002). Both are designed to perform their best with radial datasets, due to the fact that both create a mapping where each neuron captures a "pie slice" of the data according to the angular distribution of the input data (see Fig. 1). However, when SOM is trained, it approximates a Voronoi tessellation

* Corresponding author.
E-mail addresses: hector.quintian@usal.es (H. Quintián), escorchado@usal.es (E. Corchado).

http://dx.doi.org/10.1016/j.engappai.2017.01.002
Received 7 April 2016; Received in revised form 3 November 2016; Accepted 2 January 2017
Available online 12 January 2017

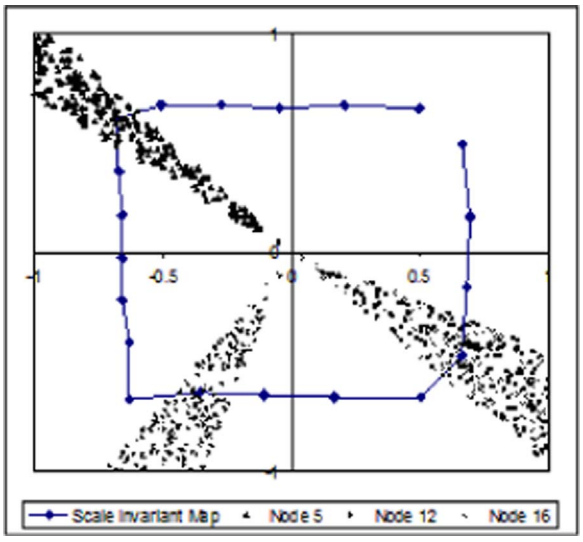

Fig. 1. Scale Invariant Map mapping, where each neuron captures a "pie slice" of the data according to the angular distribution of the input data.

of the input space (Kohonen, 1998). The Scale Invariant Map is an implementation of the negative feedback network (Fyfe, 2005) to form a topology preserving mapping. The main difference between this mapping and the SOM (Kohonen, 1998, 2013) is that this mapping is scale invariant.

Finally, another widely used clustering and classification algorithm is the Growing Neural Gas (GNG) algorithm, proposed by Fritzke (1995), Zapater et al. 2015). It is based on the Neural Gas (NG) algorithm previously proposed by Martinetz et al. (1993) for finding optimal data representations based on feature vectors, which is in turn a modification of the widely known SOM. The main characteristic of the NG algorithm is that instead of expanding through the data input space as a fixed grid of units (as done by the SOM algorithm), the NG algorithm allows the neighboring relationships of its units to change, expanding more like a gas over the data space.

The GNG method 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 (Zapater et al., 2015) is a combination of Fritzke's Growing Cell Structures (GCS) (Fritzke, 1994) and Martinetz's Competitive Hebbian learning (CHL) (Martinetz, 1993). The network topology of the GNG is generated incrementally by the CHL algorithm, which successively inserts topological connections or edges. The main principle of the CHL is that for each input x, it connects the two closest centers (measured by Euclidean distance) with an edge.

This research study presents a novel and efficient technique for data clustering called Beta-Scale Invariant Map (Beta-SIM). It is based on a modification of a topology preserving map that can be used for scale invariant classification (Baruque and Corchado, 2014; Corchado and Baruque, 2012; Baruque et al., 2011; Corchado and Colin, 2002). The main objectives of this study are:

- To study and derive a family of learning rules from Beta distribution and apply them to the Scale Invariant Map (SIM) (Baruque and Corchado, 2014, 2009) to improve the clustering and visualization of internal structure of high dimensional datasets, specifically with radial structure.
- To thoroughly study the advantages and disadvantages of the novel Beta-SIM algorithm over 2 artificial and 16 real datasets, testing its capabilities.
- To test the capacity of the novel proposed algorithm (Beta-SIM) to adapt to sparse clusters or to neglect outliers through the right combination of α and β values, depending on task to be carried out.

This paper is organized as follows: Section 2 presents in detail the SIM algorithm which leads on to the MLHL and MLHL-SIM algorithms that are explained in Sections 3 and 4. Section 5 introduces the Beta Hebbian Learning used to derive the learning rules for the new algorithm, Beta-SIM, which is described in detail in Section 6. Section 7 presents 3 quality measures, previously proposed in the literature, used to evaluate different properties of topology-preserving mapping algorithms in general. Section 8 analyzes the capabilities of the Beta-SIM algorithm by applying it to perform a detailed study over 2 artificial datasets and 16 real benchmark datasets with diverse characteristics. Finally, Section 9 contains the final conclusions and outlines future lines of research.

2. Scale Invariant Map

The main target of the family of topology preserving maps (Kohonen, 1998) is to produce low dimensional representations of high dimensional datasets, maintaining the topological features of the input space.

SIM (Baruque and Corchado, 2014, 2009) is an algorithm similar to SOM (Kohonen, 1998), but the training methodology is based on negative feedback networks (Fyfe, 2005, 1997). SIM uses a neighborhood function and competitive learning in the same way as the SOM. The SIM model is defined by Eqs. (1)–(3):

$$Feedforward: y_i = \sum_{j=1}^{N} W_{ij}x_j, \tag{1}$$

$$Feedback: e = x - W_c y_c \quad (y_c=1), \tag{2}$$

$$Weights\ update: \Delta W_i = h_{ci}\eta(x - W_c), \quad \forall i \in N_c, \tag{3}$$

where x is an N-dimensional input vector, and y an M-dimensional output vector, with W_{ij} being the weight linking input j to output i; e is the residual or error, η the learning rate, W_c refers to the weights of the winning neuron and h_{ci} represents the neighborhood function, which is a Gaussian function in this case.

The input data x_j is feedforward through weights W_{ij} to create output data y_i, where a linear summation is performed to obtain the activation of the output neurons (1). Based on the activation from the feedforward algorithm, a winner neuron is selected using the minimum Euclidean distance (the neuron whose output vector is closest to the input vector wins) or using the maximum activation (the output neuron with the highest activation wins). After selection of an output winner, denoted as c, it is deemed to be firing (y_c=1) and all other outputs are suppressed (y_i=0, $\forall i \neq c$).

The winner's activation is then used as feedback (2) using the winner's weights subtracted from the input data, and simple Hebbian learning to update the weights of all nodes in the neighborhood of the winner (3).

3. An exponential family of learning rules

Maximum Likelihood Hebbian Learning (MLHL) (Corchado et al., 2004) is a family of rules created from exponential distributions, which can be derived to express the Probability Density Function (PDF) of the residual after feedback as (4):

$$p(e) = \frac{1}{Z}exp(-|e|^p), \tag{4}$$

It can then be denoted as a general cost function associated with this network as (5):

$$J = E(-log(p(e))) = E(|e|^p + K), \tag{5}$$