Effects of Irregular Topology in Non-Planar SOM Variants

Charles R. Schmidt
MS Candidate
Department of Geography
San Diego State University

April 7, 2007

Committee Members
Dr. Sergio Rey, Geography
Dr. André Skupin, Geography
Dr. Robert Malouf, Linguistics

A Thesis Proposal presented to the faculty of San Diego State University.

Contents

1 Introduction

The use of a spherical lattice is widely suggested as a solution to the problem of the boundary effect found in the traditional euclidean Self-Organizing Map (SOM) (Ritter, 1999; Boudjemai et al., 2003; Sangole and Knopf, 2003; Wu and Takatsuka, 2006; Nishio et al., 2006). The use of the spherical lattice, however, does not completely overcome this problem. With the exception of the five platonic solids, distributing points on a sphere will always result in irregular network topology (Ritter, 1999; Harris et al., 2000). The classic method for minimizing this irregularity is to generate the spherical lattice by tessellating the sides of the icosahedron (Nishio et al., 2006). While this method will always result in a highly regular spherical topology, the main drawback is that the network size, N, grows exponentially. Other methods for arranging neurons on the sphere allow for unlimited control over network size, but yield topologies with increased irregularity (Harris et al., 2000; Wu and Takatsuka, 2005; Nishio et al., 2006). The goal of this research is to investigate the extent to which irregularity in the spherical topology effects the training of the SOM, in order to determine the suitability of irregular topologies for use in spherical SOM.

2 Research Objectives

The objective of this research is to determine the usefulness of irregular spherical topologies that offer greater control over network size. Towards that end, I will develop and test new diagnostics to measure and visualize topologically induced errors in SOM training.

- Develop Diagnostics:
  1. Visualize the reverse quantization error by mapping neurons back on to the input data
  2. Plot: internal variance of each neuron against its first-order neighborhood size
  3. Plot: internal variance of each neuron against the variance in neighborhood size for each topology
- Test Diagnostics with Various Network Topologies
  1. Generate synthetic data
  2. Train a SOM with comparable parameters for each topology

3 Background and Lit Review

The Self-Organizing Map (SOM) is an unsupervised competitive learning process developed by Teuvo Kohonen as a technique to analyze high dimensional data sets. The SOM algorithm uses an artificial neural network to organize high dimensional data onto a low dimensional lattice, or map, of neurons. Each neuron contains a reference vector that models the input data. Before training, these neuron-vectors are initialized, most commonly to random values. During the training process a randomly selected observation (input vector x) searches each neuron (reference vector m_i) to find the one to which it is most similar, referred to as its Best Matching Unit (BMU c). The BMU and its neighborhood (N_c) are then adjusted to better match that observation (Kohonen, 2000). The training process is repeated a predefined number of times, or ideally until the map converges. The traditional SOM is laid out on a two dimensional plane using either a rectangular or hexagonal topology. According to Wu and Takatsuka (2006) the hexagonal structure is more uniform and generally preferred.

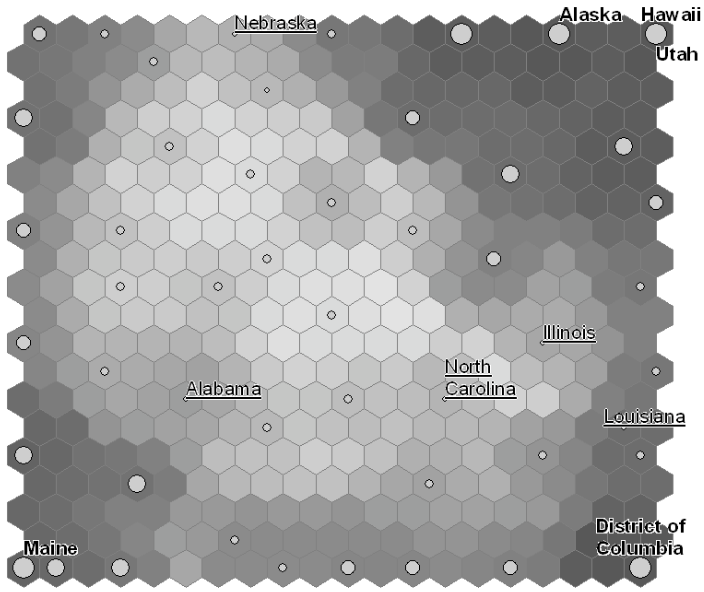

Figure 1: Fifty states plus D.C. mapped onto SOM trained with the first thirty-two census variables. Darker neurons have a relatively large differences from the mean of the states, while lighter neurons are relatively closer. Larger dots represent inputs that were poorly fit to the map, small smaller dots show inputs that were better bit to the map.

One drawback of building the neural lattice in a discrete euclidean plane is the boundary of the resulting lattice. A neuron located on the boundary has fewer neighbors and thus fewer chances of being updated (Wu and Takatsuka, 2006). As observed in Figure 1, neurons in the center of the map tend to better represent the mean of the input-space. This is arguably caused by outliers being pushed to the edges of the map, where they encounter fewer competing signals.

3.1 Spherical SOM

A way to eliminate the boundary effect is to wrap the lattice around a three dimensional object such as a sphere or torus, therefor removing the edge entirely. The toroidal SOM was introduced by Li et al. (1993), however the torus is not hugely effective for visualization, as maps generated from a torus are not very intuitive (Ito et al., 2000; Wu and Takatsuka, 2006). Ritter (1999) describes the torus as being topologically flat and suggests that a curved topology, such as that of a sphere, may better reflect directional data. A sphere also results in a more intuitive map, since we are accustomed to looking at maps based on a sphere.

Ritter (1999) first introduced the spherical SOM and several enhancements have since been suggested (Boudjemai et al., 2003; Sangole and Knopf, 2003; Nishio et al., 2006; Wu and Takatsuka, 2006). A good comparison of these enhancements can be found in (Wu and Takatsuka, 2006). All of these methods derive their spherical structure through the tessellation of a polyhedron as originally proposed by Ritter.

Wu and Takatsuka (2006) point out the importance of a uniform distribution on the sphere and that it is preferable for all neurons to have an equal number of neighbors and to be equally spaced. They find generally that the tessellation method best satisfies these conditions and specifically that the icosahedron is the best starting point (Wu and Takatsuka, 2005). Tessellation of the icosahedron results in a network of neurons, each of which have exactly six neighbors, save the original twelve which each have five. This is very close to the ideal structure in which every neuron would have exactly six neighbors. This structure has very low variances in both neuron spacing and neighbor size.

3.2 Network Size

The tessellation method used by the class of spherical SOMs based on Ritter's work results in a network size that is a function of the tessellation frequency and therefore grows exponentially. In practice 2D Euclidean SOMs also offer limited control over network size, as it is undesirable to have one dimension dramatically larger than the other. Nishio et al. (2006) try to address the issue of network size granularity by departing from the tessellation method and suggesting the use of a partitioned helix to uniformly distribute any number of neurons on a sphere. A similar method was dismissed by Wu and Takatsuka (2005) for failing to satisfy the uniformity conditions. Nishio et al. seem to have addressed variance in nearest neighbor distances, but the issue of having a uniform number of direct neighbors is still not addressed.

The literature offers little theoretical guidance on network size (Cho et al., 1996). Vesanto (2005) suggests simply using a network size of $5 * \sqrt{n}$, where n is the number of observations.

3.3 Uniformity

Wu and Takatsuka state that "[f]or SOM, it is desirable to have all neurons receive equal geometrical treatment" (Wu and Takatsuka, 2006, pp. 900). To satisfy this constraint two conditions must be met. Firstly, each neuron should occupy the same amount of space on the given surface. Secondly, each neuron should be bordered by the same number of surrounding neurons, and we should maximize that number. The first condition is hugely important for visualization, but irrelevant for training. During the training of the SOM only the topology of the neurons is considered.

Based solely on measures of neuron spacing Wu and Takatsuka (2005) dismissed a method proposed by Rakhmanov et al. (1994) for distributing points on a sphere. Similarly Nishio et al. (2006) use these variance measures to support their helix algorithm for distributing points on a sphere. Table 1 shows that these metrics can be misleading and may not be comparable across topologies. The traditional rectangular and hexagonal topologies have no variance in neuron spacing and the generally preferred hexagonal structure displays greater variance in neighborhood size than the rectangular structure. The torus by comparison would have variance in neuron spacing, yet no variance in neighborhood size. The distance between two neurons is only considered during the formation of the neuronal network. At this stage the spacing is significant as it plays a part in determining neuron adjacency. However using this measure to evaluate potential topologies for use in SOM may be misleading.

Methods, for distributing points on the sphere, which allow for fine grained control over network size produce slightly more irregular topologies. However, no discussion of these irregularities or their effects on SOM training has occurred in the literature. Given that limited theoretical guidance is available for choosing network size the desire for finer control over the network size, should not be overlooked. In particular for a larger SOM the ideal network size may not be achievable via tessellation of the icosahedron.

Table 1: Variances in Topologies

Topology	Grid Size	Neuron Spacing	Variance in Neighborhood Size
Rectangular	9x18	1	0.2716
Hexagonal	9x18	1	1.2138
Tessellation	162	0.25319 - 0.31287	0.0686
Rakhmanov	162	0.15779 - 0.30069	0.2908

4 Data and Methodology

4.1 Diagnostics

4.1.1 Visualize reverse mapping

The quantization error (QError) of a SOM is a simple way to measure how well data matches a trained SOM. The QError is measured by averaging the euclidean distance of each observation and its best match. The QError can be decomposed to show how well each observation fits the SOM. For the first diagnostic tool I will find the euclidean distance between each neuron and its best matching observation. The result will be a map that shows how well the neurons fit the data. This can be extended to produce a map that shows the distances of neurons to the mean (or median) of the input data.

4.1.2 Plot the internal variance of each neuron against its first-order neighborhood size

In tradition SOM outlying observations are pushed to the edge of the map where they encounter fewer competing signals. A prime example of this is the "Utah-Hawaii" case shown in figure 1. Relying only on the SOM one would be left to believe that the two states are similar. At closer look we see that the QError from Utah to the neuron is 1.509, the QError from Hawaii to the neuron in 1.505 and the QError from Utah to Hawaii is 3.014. In this case only Utah and Hawaii were mapped to that neuron. In a case where multiple observations land on the same neuron it is possible to measure average pairwise QErrors between those observation. This gives us a notion of internal variance for each neuron with multiple observations. It would be expected that in traditional SOM neurons closer to the edge will have higher internal variances. This can be extended to spherical SOM be looking at the degree of each neuron ($deg(m_i)$ or the number of adjacent neurons).

The first-order adjacency (A) matrix for a given network topology will be utilized in the next two diagnostics. The column sums of A tell use the size of each neurons first-order neighborhood. A SOM with completely regular network topology (i.e. the torus) will have no variance in these column sums. For irregular networks it will be useful the visualize these variances as a map. This will show us where topologically induced errors are occurring.

Once the SOM is trained

4.1.3 Plot the internal variance of each neuron against the variance in neighborhood size for each topology

4.2 Empirical Analysis

4.2.1 Synthetic Data

4.2.2 Train SOM

4.2.3 Apply Diagnostics

The cost of relying on Ritter's tessellation method is decreased control over network size. Ritter's tessellation methods results in a network size that grows at a rate of $N = 2 + 10 * 4^f$, where f is the frequency of tessellation. Wu and Takatsuka (2006) offer a slight improvement. Rather than recursively subdividing the faces, they redivide the original icosahedron with each step resulting in $N = 2 + 10 * f^2$.

It is useful to represent our neural lattice as a graph, G. We know that any arrangement of neurons onto the surface of a sphere will as such the network can be effectively treated as a graph. Doing so enables us to use the properties of the graph and the principles of graph theory to help us understand the relationship between neurons during training.

the variance in neuron spacing should be minimized. The first condition ensures that each neuron will occupy is only of concern when visualizing the SOM.

The get at the second condition we must first define the degree of a neuron, deg(n) to be number of its direct neighbors. Whit that in mind the second condition is that the variance in the degree of the neurons must also be minimized.

This statement may be somewhat misleading to those investigating alternative

For the purpose of SOM visualization it is important for each neuron to receive equal geometrical treatment.

It is useful to represent the neural network as a graph, G, in order use the... In order to examine the irregularities in the neural network it is useful ....

The basic problem of the "boundary effect" is that neurons on the edge have fewer neighbors. Yet there are only five possible arrangements of points on a sphere such that all points have the same number of neighbors. Any spherical lattice consisting of more then twenty (dodecahedron) neurons will contain topological irregularities. This is to say that not all neurons will have the same number of neighbors. The importance of these irregularities and the magnitude of their effects on SOM training is not known. This goal of this research is to determine whether more flexible network structures may be used in spherical in SOM without introducing significant errors. To accomplish this goal, basic methods in network analysis with be combined with the result from several empirical training runs each utilized different topology.

4.3 Synthetic Data

In order to have a basis for comparison

References

Boudjemai, F., Enberg, P. B., and Postaire, J. G. (2003). Surface modeling by using self organizing maps of kohonen. In *Systems, Man and Cybernetics, 2003. IEEE International Conference on*, volume 3, pages 2418–2423 vol.3.

Cho, S., Jang, M., and Reggia, J. A. (1996). Effects of varying parameters on properties of self-organizing feature maps. *Neural Processing Letters*, V4(1):53–59.

Harris, J. M., Hirst, J. L., and Mossinghoff, M. J. (2000). *Combinatorics and graph theory*. Springer, New York.

Ito, M., Miyoshi, T., and Masuyama, H. (2000). The characteristics of the torus self organizing map. In *Proceedings of 6th International Conference ON Soft Computing (IIZUKA2000)*, volume A-7-2, pages pp.239–244, Iizuka, Fukuoka, Japan.

Kohonen, T. (2000). *Self-Organizing Maps*. Springer, 3 edition.

Li, X., Gasteiger, J., and Zupan, J. (1993). On the topology distortion in self-organizing feature maps. *Biological Cybernetics*, V70(2):189–198.

Nishio, H., Altaf-Ul-Amin, M., Kurokawa, K., and Kanaya, S. (2006). Spherical som and arrangment of neurons using helix on sphere. *IPSJ Digital Courier*, 2:133–137.

Rakhmanov, E. A., Saff, E. B., and Zhou, Y. M. (1994). Minimal discrete energy on the sphere. *Mathematical Research Letters*, 1:647–662.

Ritter, H. (1999). Self-organizing maps on non-euclidean spaces. In Oja, E. & Kaski, S., editor, *Kohonen Maps*, pages 97–110. Elsevier, Amsterdam.

Sangole, A. and Knopf, G. K. (2003). Visualization of randomly ordered numeric data sets using spherical self-organizing feature maps. *Computers & Graphics*, 27(6):963–976.

Vesanto, J. (2005). Som toolbox: implementation of the algorithm.

Wu, Y. and Takatsuka, M. (2005). Geodesic self-organizing map. In Erbacher, R. F., Roberts, J. C., Grohn, M. T., and Borner, K., editors, *Proc. SPIE Vol. 5669*, volume 5669 of *Visualization and Data Analysis 2005*, pages 21–30. SPIE.

Wu, Y. and Takatsuka, M. (2006). Spherical self-organizing map using efficient indexed geodesic data structure. *Neural Networks*, 19(6-7):900–910.