Space-time trajectories in a planar and torus-based self-organising map: the importance of eliminating boundary effects

Nick Mount and Dan Weaver

School of Geography, University of Nottingham, Nottingham NG7 2RD
Tel. +44(0)115 951 5428 Fax. +44 (0)115 951 5249
nick.mount@nottingham.ac.uk danofweaver@googlemail..com

KEYWORDS: SOM, self-organizing map, torus, projection, space-time, boundary effects

1. Self-organizing maps and boundary effects

Traditional self-organizing maps (SOMs) are a multi-dimensional clustering method, based on an unsupervised neural network, which maps complex input data to a two dimensional neuron array. Mapping is achieved using a competitive, iterative algorithm, which maps sample input data to individual neurons in the array based on the fit between the multi-dimensional value set (i.e. an n-dimensional vector) in the sample data and the corresponding vectors associated with each neuron (usually using Euclidean distance as the parameter by which fit is assessed). In each iteration, the best matching neuron (BMN) is adjusted to become more like the sample data, with proximal neurons similarly adjusted, though limited by a decay function so that the magnitude of adjustment decreases with increasing distance from the BMN. A Gaussian decay function is used as standard, with the amplitude and spread of the function declining progressively with the number of iterations; so allowing increasingly fine adjustment of the neuron vectors to match the input data.

Given the two-dimensional nature of the neuron array, a two-dimensional Gaussian decay function is required (figure 1) which can is computed using the following formula:

$$f(x,y) = A\exp-\left(\frac{(x-x_0)^2}{2\sigma_x^2}\right)-\left(\frac{(y-y_0)^2}{2\sigma_y^2}\right)$$

where A is the amplitude, x_0 , y_0 is the centre, and σ_x and σ_y are the spreads of the function.

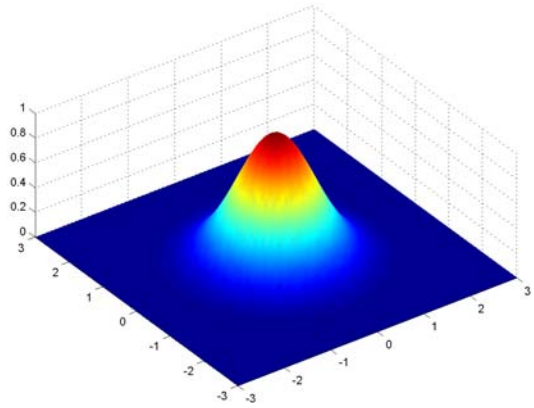

Figure 1. The form of a two-dimensional Gaussian decay function.

The finite nature of the neuron array in the traditional SOM means that neurons close to the boundary have truncated neighbourhoods, with the truncation being greatest in the corners of the neuron array. Indeed, the magnitude of the boundary effect is proportional to the number of neurons in the array and the spread of the decay function. Consequently, when the decay function is centred on boundary neurons, a smaller number of neurons are adjusted relative to decay function centred on a neuron in the middle of the array (see figure 2). Kiang et al., (1997) show that it is possible for up to 75% of the possible adjustments to be lost due to the array boundary.

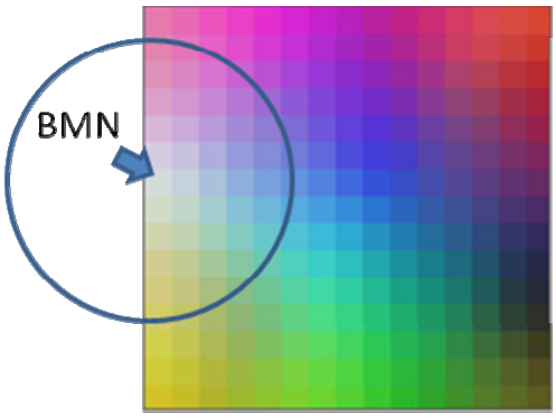

Figure 2. A 15 x 15 SOM with the location of the BMN shown together with the limit of the decay function. The boundary location of the BMU means that, compared to a BMN located in the middle of the SOM, half the number of neurons are located under, and adjusted by the decay function.

The impact of this truncation is a reduction in the performance of the SOM, with divided clusters common, particularly at opposite boundaries and corners of the SOM. The impact can be shown by mapping a colour palette (figure 3) containing the full range of RGB values to a SOM array. A correctly functioning SOM will maintain the proximity relations of the colours within the palette in the mapped output. However, figure 3 shows the impact of the SOM boundaries, which have caused two separate clusters for red-dominant and blue-dominant hues, incorrectly mapping the proximity relations in the input image.

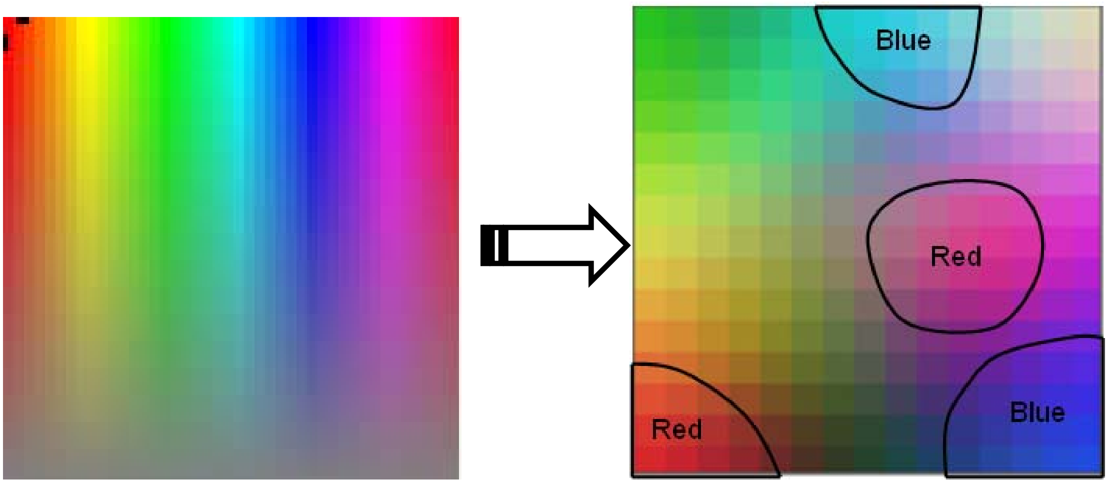

Figure 3. A SOM used to cluster red, green and blue colour bands from a palette. Boundary effects have caused inconsistencies in the SOM's proximity relationships.

2. Boundary effects in spatio-temporal self-organizing maps

Boundary effects become particularly problematic in spatio-temporal self-organizing maps (e.g. Weaver and Mount, 2007). Here the data used to train the SOM are sampled across multi-temporal input data with the weight vectors against which the SOM learns implicitly including a temporal dimension; resulting in the evolution of a spatio-temporal SOM (Kohonen, 1995). SOM locations to which input data for each time period have been mapped can then be linked together so that a set of vectors describing the space-time trajectory can be extracted (figure 4). These vectors represent the changes in the multiple dimensions of each sampled datum through time, with high-magnitude vectors indicating large-scale changes, and low-magnitude vectors indicating only minor change. By comparing the set of vectors associated with a data set, the SOM can then be used to differentiate processes encoded within the data (e.g. Skupin and Hagelman).

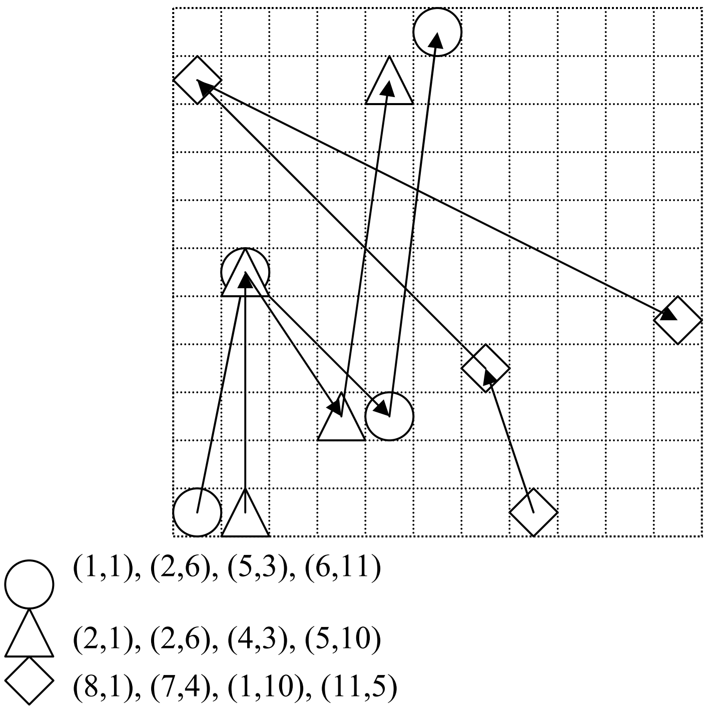

Figure 4. Space time trajectories for three samples in an 11x11 spatio-temporal SOM. The

data represented by the circle and triangle vary similarly through time. The datum represented by the diamond has a very different response in its parameters through time.

The divided clusters caused by SOM boundaries will increase the magnitude of the vectors starting and / or ending in the divided clusters; erroneously suggesting high-magnitude processes where none exist and making impossible to compare them with other vectors. As the SOM process is dependent on a random-sampling strategy, it is impossible to identify which trajectories will be affected *a priori*, making it impossible to identify and remove problem trajectories as a post-processing operation. Given the importance of boundary effects on space-time trajectories, it is surprising that examples in the geographical literature (e.g. Skupin and Hagleman, 2005) fail to explicitly recognise or address the problem.

3. Solving spatio-temporal SOM boundary problems

The solution to the boundary problem requires the removal of the boundaries via the projection of the SOM onto a three-dimensional surface. A torus (figure 5) offers a relatively simple projection as it is, in effect, a dual-axis wrapping of the planar SOM array, which removes all boundaries by connecting opposing boundaries. From a programming perspective, it requires computation of the complete set of *x* and *y* coordinates of neurons to be modified around the BMN for each iteration, irrespective of whether these coordinates actually exist within the SOM array, and knowledge of the extent to which each *x* and *y* coordinate exceeds the SOM limits.

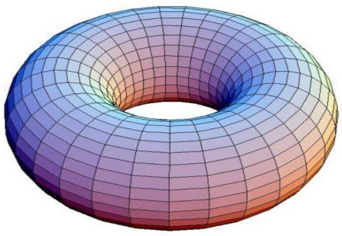

Figure 5. A torus-based projection of a planar surface produced by a dual axis wrapping of the planar surface.

For example, consider the case of a 10 x 10 SOM array, an iteration in which a BMN neuron is found to be located at location (10,5) and a decay function spread that requires adjustment of the surrounding 8 neighbours to the BMN (see figure 6). Neurons existing at (11,4), (11,5) and (11,6) should be adjusted but these exceed the SOM array extent in the *x*-axis by 1. Therefore, neurons at (1,4), (1,5) and (1,6) are adjusted, thereby wrapping the planar SOM around its *y*-axis.

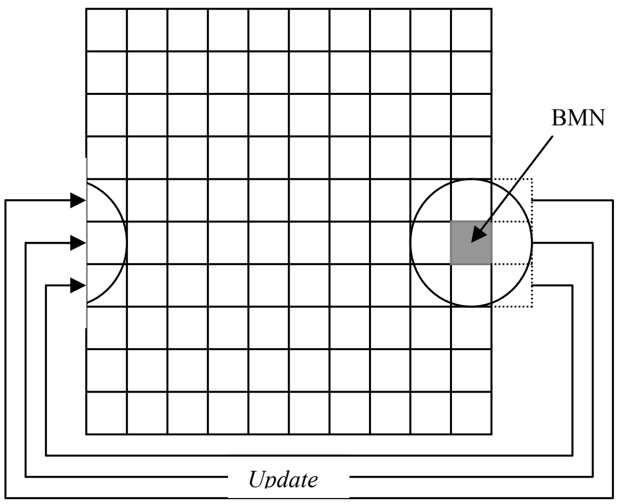

Figure 6. Creating a torus-based SOM by wrapping axes.

4. An experiment to exemplify the impacts on space-time trajectories of a torus-based and planar SOM.

The geographically-integrated, spatio-temporal, self-organising map (GISTSOM) software (Weaver and Mount, 2007) being developed in the School of Geography at the University of Nottingham offers the ability to compare space-time trajectories for planar and torus-based SOMs, identifying erroneous trajectories created as a result of boundary effects.

To investigate the impact of SOM boundaries on space-time trajectories, GISTSOM has been used to assess the space-time trajectories in two RGB images, derived from a colour palette (figure 7). Palette(0) is modified to palette(1) via the application of a 'swirl' effect, with its origin on the image midpoint. As a result, the difference in the RGB values in pixels close to the centre of the two images (image coordinate (149,138)) will show little variability in their RGB values across between palette(0) and palette(1), whereas those pixels further away from the origin will show greater variability. When mapped to a spatio-temporal SOM, pixels in the central portion of the image should have low-magnitude trajectories, whist those away from the centre should have higher-magnitude trajectories.

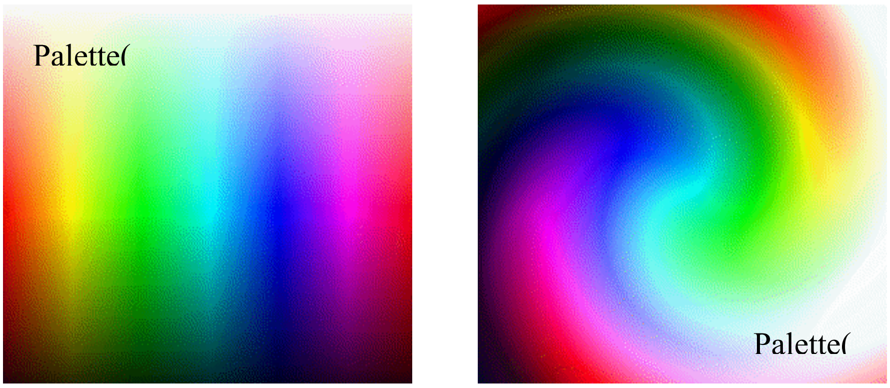

Figure 7. The two palettes using in the SOM trajectory comparison. Palette(1) is identical to palette(0) except for the fact that a swirl effect has been applied with its origin at the image centre point.

The input images were mapped to both a planar and a torus-based 15x15 SOM. In both cases a Gaussian decay function with a start variance of 13 and a linear decay / iteration relationship in both the amplitude and spread parameters was used. The SOM was trained using 2000 iterations. To exemplify the impact of the boundaries, the trajectory lengths associated with 10 pixels at increasing distance from the image centre were computed for both SOMs (figure 8).

Given the nature of the input data, a correctly functioning SOM will show a monotonic pattern of increasing trajectory length with distance from the image centre, with the rate of trajectory length change being approximately stable. No large, sudden increases in trajectory length should occur. However, it is clear from figure 8 that, whilst the torus-based SOM is able to replicate this ideal, the planar SOM is unable to reproduce this pattern, being neither monotonic nor stable in the rate of trajectory change. This is due to a boundary effect associated with the sample at 65 pixels from the image centre. The boundary forces a long trajectory at this sample point, and those at 78 and 91 pixels, causing a rapid increase in the trajectory length and preventing its monotonic increase.

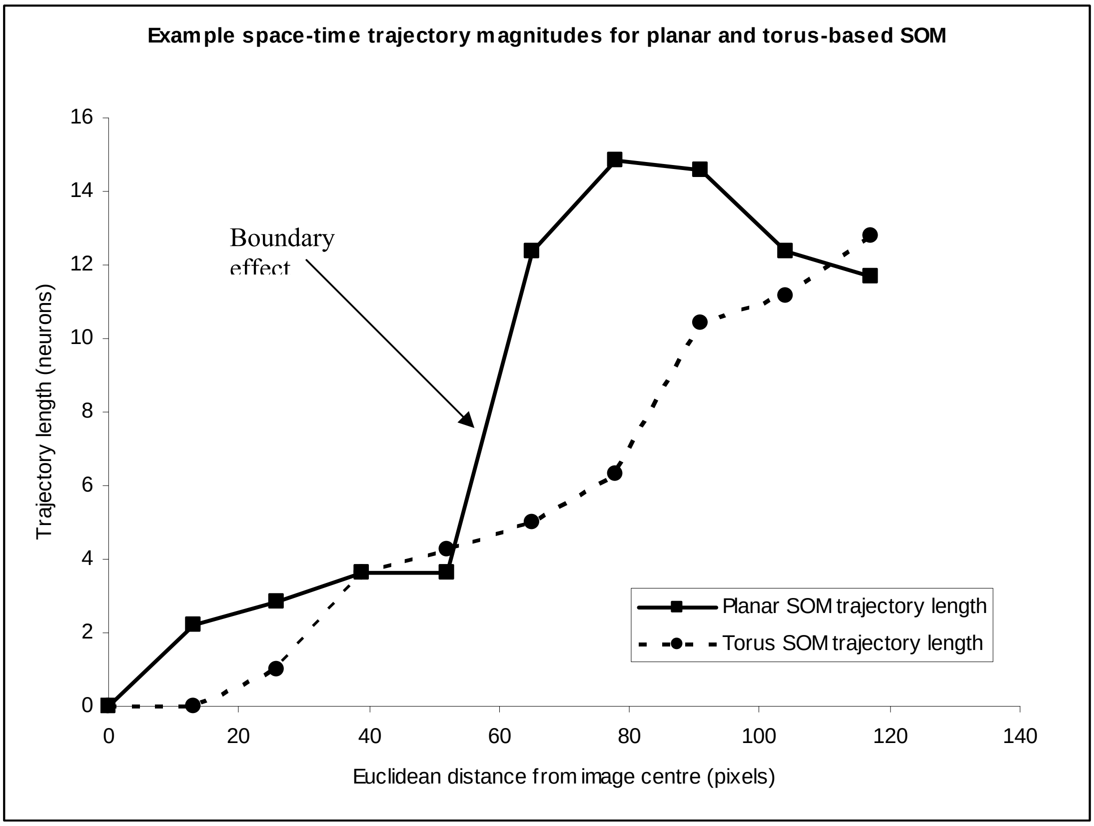

Figure 8. A comparison of the trajectory lengths associated with 11 sample data points in a planar and torus-based SOM with the impact of the boundary effect highlighted.

5. Conclusions

The example analysis presented here demonstrates the impact of planar SOM boundaries on trajectory lengths. Planar SOMs result in increased trajectory lengths where neuron clusters are split across the boundaries, erroneously indicating high-magnitude changes in the space-time structure of the input data where none exists. Where SOMs are used to elucidate space-time trajectories, a projected SOM must be used, with a torus offering a relatively simple and effective projection.

6. References

Kiang, M.Y., Kulkarni, U.R., Goul, M., Philippakis, A., Chi, R.T., Turban, E. (1997). Improving the effectiveness of self-organizing map networks using a circular Kohonen layer. *Proc. 13th Hawwaii Int. Conf. on System Sciences*, 521-529.

Kohonen, T. (1995). Self-organizing maps. Springer, Berlin.

Skupin, A., Hagelman, R. (2005). Visualizing demographic traectories with self-organizing maps. *Geoinformatica*, **9**(2), 159-179.

Weaver, D., Mount, N.J. (2007). The use of Kohonen mapping for the elucidation of space-time trajectories of multiple parameters: potential applications in fluvial geomorphology. *Proceedings of the GIScience Research UK 15th Annual Conference, University of Maynooth*.

Biographies

Nick Mount is a lecturer in GISc at the University of Nottingham and Dan Weaver completed Nottingham's MSc in GISc in 2006 and is now a Research Associate in GISc. The GISTSOM software is being developed with the aid of a New Researcher Fund grant and is due for Beta release in June 2008.