The Topographic Neural Gas

Marian Peña and Colin Fyfe

University of Paisley, Paisley PA1 2BE, Scotland,
marian.pena@paisley.ac.uk
colin.fyfe@paisley.ac.uk

Abstract. We have recently investigated a family of algorithms which use the underlying latent space model developed for the Generative Topographic mapping(GTM) but which train the parameters in a different manner. Our first model was the Topographic Product of Experts (ToPoE) which is fast but not so data-driven as our second model, the Harmonic Topographic Mapping (HaToM). However the HaToM is much slower to train than the ToPoE. In this paper we introduce ideas from the Neural Gas algorithm to this underlying model and show that the resulting algorithm has faster convergence while retaining the good quantization properties of the HaToM.

1 Introduction

Clustering is one of the fundamental problems in data mining. There are different techniques, the most popular being K-means and its harmonic variant K-Harmonic Means [11]. These are sometimes combined with topology-preserving algorithms such as Neural Gas (NG)[6], the Self-organizing Map (SOM)[5] and the Generative Topographic Mapping (GTM)[1] which tend to be used for visualisation of datasets. The neighborhood cooperation in these algorithms also reduces the influence of initialisation [2]. The topology preservation however may limit the efficiency of clustering due to the fixed topology in the algorithm.

One drawback of K-Means is its sensitivity to initialisation of the centres, that can lead to convergence to a local minima. The SOM algorithm can be considered as a topology preserving mapping generalisation of K-Means. We[4][9][10] have recently investigated a family of algorithms which use the underlying latent space model developed for the Generative Topographic Mapping (GTM) but which train the parameters in a different manner. Our first model was the Topographic Product of Experts (ToPoE).

K-Harmonic Means overcomes the initialisation problem by using harmonic means instead of arithmetic means. Recently we have used this clustering technique in a topology preserving map called The Harmonic Topographic Mapping (HaToM) that shares a common structure with the GTM map, but the centres are organised by K-Harmonic Means.

We have shown that HaToM is more responsive to the data than ToPoE but this comes at a cost of an increase in computation time. In this paper, we introduce ideas from the Neural Gas algorithm and show that the resulting method retains the good quantization properties of HaToM but is much faster.

2 GTM, ToPoE and HaToM

The GTM[1] was introduced as a principled alternative to Kohonen's SOM[5]. It begins with a fixed set of points, $t_k, k = 1, ..., K$, in latent space which have some regular topology, such as lying on a grid. These latent points are then mapped through a set of nonlinear basis functions, tipically Gaussians, to an intermediate feature space which is then mapped to a set of centres, m_k, in data space. This last mapping is a linear mapping with a set of parameters, W, which are updated by treating the complete mapping as a mixture of experts. [1] uses the Expectation Maximization (EM) algorithm to train the parameters such that the m_k lie on the data manifold and so, by investigating the responsibilities that each latent point has for each data point, the resulting mapping can be used to visualize the data.

ToPoE[4] uses the same underlying mapping, $t_k \rightarrow m_K$, but treats the structure as a product of experts. Training is done by gradient descent on the resulting mapping. HaToM[9, 10] again uses this underlying mapping but uses K-Harmonic Means[11] to train the parameters. We have previously shown that this algorithm is more data driven than ToPoE but this comes at increased computational expense.

3 Neural Gas

Vector quantization methods encode a set of data points in n-dimensional space with a smaller set of reference vectors m_k, $k = 1, ..., N$. The m_k are determined such that the expected Euclidean distance between all data vectors and their corresponding reference vectors becomes minimal. Neural Gas [6] is a vector quantization technique with soft competition between the units. In each training step, the squared Euclidean distances

$$d_{ik} = \|\mathbf{x}_i - \mathbf{m}_k\| = (\mathbf{x}_i - \mathbf{m}_k)^T * (\mathbf{x}_i - \mathbf{m}_k) \tag{1}$$

between a randomly selected input vector x_i from the training set and all reference vectors m_k are computed; the vector of these distances is d. Each centre k is assigned a rank $r_k(d) = 0, ..., N - 1$, where a rank of 0 indicates the closest and a rank of N-1 the most distant centre to x. The learning rule is then

$$m_k = m_k + \varepsilon * h_\rho[r_k(d)] * (x - m_k) \tag{2}$$

The function

$$h_\rho(r) = e^{(-r/\rho)} \tag{3}$$

is a monotonically decreasing function of the ranking that adapts all the centres, with a factor exponentially decreasing with their rank. The width of this influence is determined by the neighborhood range ρ. The learning rule is also affected by a global learning rate ε. The values of ρ and ε decrease exponentially from an initial positive value ($\rho(0)$, $\varepsilon(0)$) to a smaller final positive value ($\rho(T)$, $\varepsilon(T)$) according to

$$\rho(t) = \rho(0) * [\rho(T)/\rho(0)]^{(t/T)} \tag{4}$$

and

$$\varepsilon(t) = \varepsilon(0) * [\varepsilon(T)/\varepsilon(0)]^{(t/T)} \tag{5}$$

where t is the time step and T the total number of training steps, forcing more local changes with time.

There is also a Growing version of Neural Gas[3] that learns the topology of the data by combining NG with Competitive Hebbian Learning (CHL), which is then closer to the SOM algorithm. In our algorithm Neural Gas is embeded in a GTM-like structure.

4 Topographic Neural Gas

Topographic Neural Gas (ToNeGas) unifies the underlying structure in GTM for topology preservation, with the technique of Neural Gas. We thus have a number of latent points (organised in a two dimensional grid as in the SOM algorithm), that are mapped to a feature space by M Gaussian functions, and then into the data space by a matrix W. Each latent point, indexed by k is mapped, through a set of M basis functions, $\Phi_1(), \Phi_2(), \cdots, \Phi_M()$ to a centre in data space, $\mathbf{m}_k = \Phi(t_k) * W$. The centres in data space are then clustered using the NG algorithm. The algorithm has been implemented based on the Neural Gas algorithm code included in the SOM Toolbox for Matlab [8].

The steps of the algorithm are as follows:

1. Initialise K to 2. Initialise the W weights randomly and spread the centres of the M basis functions uniformly in latent space.
2. Initialise the K latent points uniformly in latent space. Set count=0.
3. Calculate the projection of the latent points to data space. This gives the K centres, $\mathbf{m}_k = \Phi(t_k)^T * W$.
4. Select randomly a datapoint
5. Calculate the distances between the datapoint selected and all the centres
6. Calculate the rank of each centre depending of the previous distance, and the neighborhood function $h_\rho(r) = e^{(-r/\rho)}$
7. Recalculate centres using the learning rule $\mathbf{m}_k = \mathbf{m}_k + \varepsilon * h_\rho[r_k(d)] * (\mathbf{x} - \mathbf{m}_k)$
8. If count<MAXCOUNT, count= count +1 and return to 4
9. Recalculate W using

$$W = \begin{cases} (\Phi^T\Phi + \delta I)^{-1}\Phi^T\mathbf{\Xi} & \text{if } K < M \\ (\Phi^T\Phi)^{-1}\Phi^T\mathbf{\Xi} & \text{if } K \geq M \end{cases}$$

10. If $K < K_{max}$, $K = K + \mathit{increment}$ and return to 2.
11. For every data point, $\mathbf{x}_i$, calculate the Euclidean distance between the i^{th} data point and the k^{th} centre as $d_{ik} = ||\mathbf{x}_i - \mathbf{m}_k||$.

12. Calculate responsibilities that the k^{th} latent point has for the i^{th} data point and the projections of each datapoint in latent space

$$r_{nk} = \frac{C_\lambda(n,k)}{\sum_{j=1}^{K} C_\lambda(n,j)} \text{ and } y_n = \sum_{k=1}^{K} r_{nk} t_k \tag{6}$$

where t_k is the position of the k^{th} latent point in latent space, and $C_\lambda(n,k)$ the tri-cube Kernel

$$C_\lambda(n,k) = D\left(\frac{|\mathbf{x}_n - \mathbf{m}_k|}{\lambda}\right) \text{ where } D(t) = \begin{cases} \frac{3}{4}(1-t^2) & \text{if } |t| < 1 \\ 0 & \text{otherwise} \end{cases} \tag{7}$$

We have used this growing method with HaToM but have found with the addition of the NG learning, we can increment the number of latent points by e.g. 10 each time we augment the map. With HaToM, the increase can only be one at a time to get a valid mapping. The visualisation is provided by the projection of each datapoint to latent space y_n, using the responsibilities of all the centres for each data point r_{nk}, and the fixed centres in latent space t_k. The responsibilities include the tri-cube Kernel that proved to be better also for HaToM[10]. One of the advantages of this algorithm is that the Neural Gas part is independent of the non-linear projection, thus the clustering efficiency is not limited by the topology preservation restriction.

5 Simulations

We apply this new algorithm to a real dataset of 18 dimensions, and two artificial datasets from the Fundamental Clustering Problems Suite that are complicated to cluster for different reasons.

5.1 The Algae data set

This is a set of 118 samples from a scientific study of various forms of algae some of which have been manually identified. Each sample is recorded as an 18 dimensional vector representing the magnitudes of various pigments. 72 samples have been identified as belonging to specific classes of algae which are labeled from 1 to 9. 46 samples have yet to be classified and these are labeled 0. ToNeGas is able to cluster this data correctly (Figure 1). In this case we used wider responsibilities to spread the clusters, but as with HaToM, the projection depicts tighter clusters with narrower responsibilities.

5.2 The Hepta and Target dataset

We use two of the datasets that appear in The Fundamental Clustering Problems Suite (FCPS)[1]. We use specifically the Hepta and the Target algorithm; the first

[1] http://www.mathematik.uni-marburg.de/ databionics/

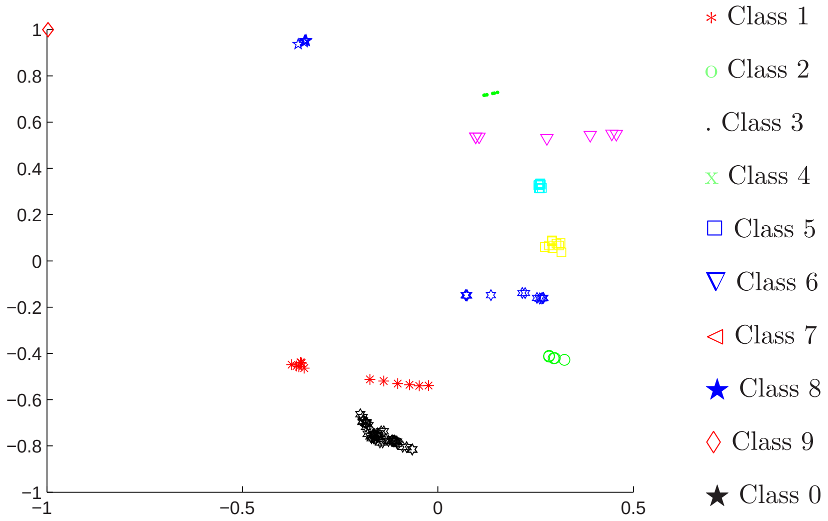

Fig. 1. ToNeGas projection of the 9 labelled algae classes and 1 unlabelled class (0).

one has clusters with different densities while the second one includes several outliers.

For both datasets (Figure 2 and Figure 3) the Topographic Neural Gas separates well the clusters, projecting the right topology into the latent space. The centres (bottom left of the Figures) are mainly located within the clusters.

The Harmonic Topographic Mapping proved to be good as well in separating these datasets (see [9] and [10]. To illustrate how the clustering speed of NG makes a great improvement of ToNeGas over HaToM we evaluate the time convergence for both algorithms and four datasets in Table 1. The difference in time is noticeable, specially when the number of datapoints is large.

Table 1. Convergence time (seconds) for HaToM and ToNeGas.

Dataset	Four clusters	Algae	Hepta	Target
No points	800	118	212	770
Dim	2	19	3	2
HaToM	174.47	7.07	17.19	155.19
ToNeGas	20.21	6.10	7.24	19.23

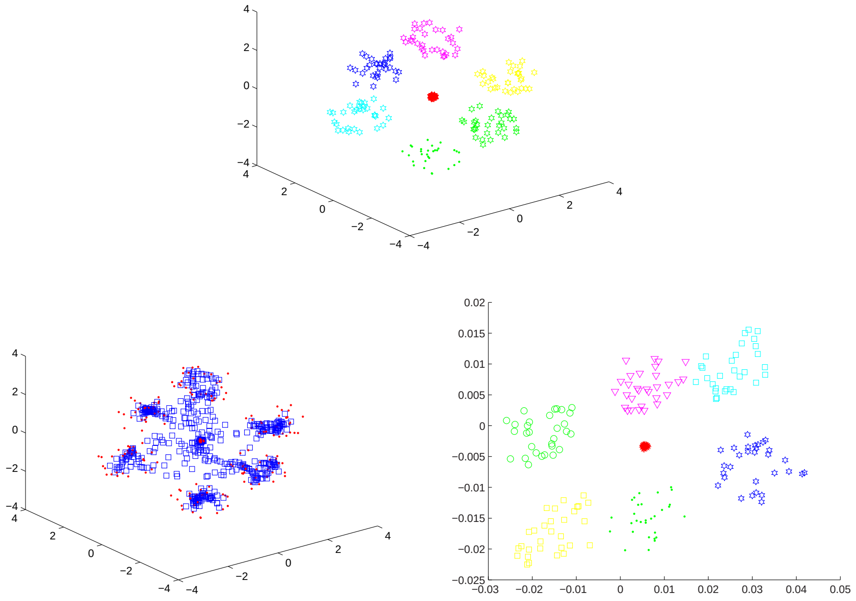

Fig. 2. Original data (top), □ centres in data space (bottom left) and ToNeGas projection of the hepta data (bottom right).

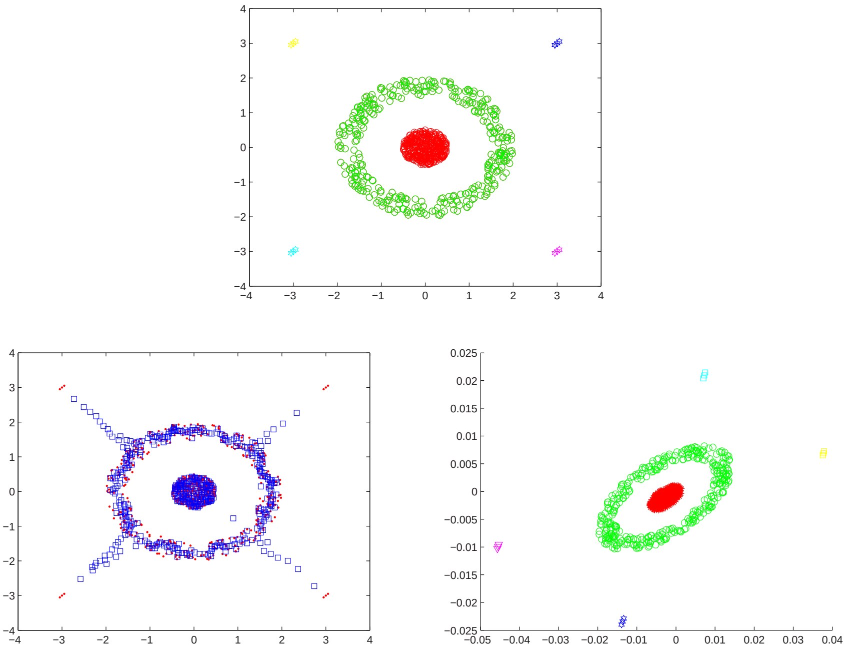

Fig. 3. Original data (top), □ centres in data space (bottom left), and ToNeGas projection of the target data (bottom right).

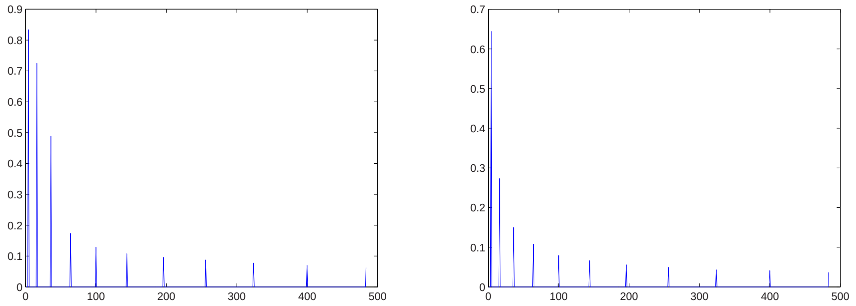

Fig. 4. Mean quantisation error over time for the Harmonic Topographic Mapping (left) and the Topographic Neural Gas (right).

Another possible criterion for comparison is the reduction in the Mean Quantisation error (MQE) while growing the map. In this experiment we calculate the MQE every time we add new latent points to the map, that is after finishing each run of the clustering technique (K-Harmonic Means for HaToM and Neural Gas for ToNeGas). We can see in Figure 4 that both techniques reduce the MQE, but the change is much more remarkable for ToNeGas.

6 Conclusions

We have presented a new algorithm for vector quantization and visualisation that integrates the Neural Gas and the underlying structure of the GTM algorithm. The clustering speed of Neural Gas gives an important improvement over the previously developed algorithm, the Harmonic Topographic Mapping, and has also proved to reduce the mean quantisation error much more than the latter.

The Topographic Neural Gas gains advantages from the Neural gas clustering as well as from the GTM like structure. Three main advantages of NG model are [6]: (1) faster convergence to low distortion errors, (2) lower distortion error than that resulting from K-means clustering, maximum-entropy clustering and Kohonens self-organizing map algorithm [5], obeying a stochastic gradient descent on an explicit energy surface. From the non-linear projection from latent space to data space, the algorithm obtains topology preservation as well as a visualisation application in a low dimensional grid.

References

1. Bishop, C. M. and Svensen, M. and Williams, C. K. I.: GTM: The Generative Topographic Mapping. Neural Computation (1997)
2. Cottrell,M and Hammer,B and Hasenfu, A and Villmann, T. Batch neural gas. WSOM 2005.
3. Fritzke, F.A Growing Neural Gas Network Learns Topologies. Advances in Neural Information Processing Systems 7 (NIPS'94), pages 625–632, Cambridge, 1995. MIT Press.
4. Fyfe, C. Two topographic maps for data visualization, Data Mining and Knowledge Discovery, 2006.
5. Kohonen, T. Self-Organization and Associative Memory. Springer-Verlag (1984)
6. Martinetz,T.M. and Berkovich, S.G. and Schulten, K.J. 'Neural-gas' network for vector quantization and its application to time-series prediction. IEEE Transactions on Neural Networks. **4** Volume 4 (1993) 558–569
7. Martinetz, Th. and Schulten, K. Topology representing networks. Neural Networks, 7 (1994) 507522
8. Neural Networks Research Centre, Helsinki University of Technology, SOM Toolbox, http://www.cis.hut.fi/projects/somtoolbox/
9. Peña,M. and Fyfe, C.: Model- and Data-driven Harmonic Topographic Maps. WSEAS Transactions on Computers **4** Volume 9 (2005) 1033-1044
10. Peña, M. and Fyfe, C.: Outlier Identification with the Harmonic Topographic Mapping. 14 th European Symposium on Artificial Neural Networks , ESANN (2006)
11. Zhang, B.:Generalized K-Harmonic Means – Boosting in Unsupervised Learning. Tech. report. HP Laboratories, Palo Alto. (2000)