Juha Vesanto and Mika Sulkava (2002). Distance matrix based clustering of the Self-Organizing Map. In Dorronsoro, J. R., editor, Proceedings of the 12th International Conference on Artificial Neural Networks (ICANN 2002). Madrid, Spain, 27-30 August 2002. Lecture Notes in Computer Science, volume 2415, pages 951-956.

Distance matrix based clustering of the Self-Organizing Map

Juha Vesanto and Mika Sulkava

Neural Networks Research Centre
Helsinki University of Technology
P.O.Box 5400, FIN-02015 HUT, Finland
Juha.Vesanto@hut.fi, Mika.Sulkava@hut.fi

Abstract. Clustering of data is one of the main applications of the Self-Organizing Map (SOM). U-matrix is a commonly used technique to cluster the SOM visually. However, in order to be really useful, clustering needs to be an automated process. There are several techniques which can be used to cluster the SOM autonomously, but the results they provide do not follow the results of U-matrix very well. In this paper, a clustering approach based on distance matrices is introduced which produces results which are very similar to the U-matrix. It is compared to other SOM-based clustering approaches.

1 Introduction

The Self-Organizing Map (SOM) [2] quantizes the training data set with a representative set of prototype vectors. The quantization process is regularized by the neighborhood relations defined between map units so that the topology of the data set is preserved. These properties make the SOM an important tool in data mining with main applications in visualization and clustering [8].

One of the most frequently used methods in clustering using SOM is the U-matrix [6] which visualizes distances between each map unit and its neighbors (see Fig. 1b). Unfortunately, when clusters are identified by visual inspection only, the procedure requires human intervention, and the results may be different when performed by different people. In this paper, automated clustering of the SOM based on the distances between neighbors is investigated. The introduced algorithm is based on the one proposed by Vellido *et al.* [7]. It is significantly improved by additional constraints and a pruning phase.

2 Base clusters

The SOM consists of a regular, usually two-dimensional, grid of map units. Each unit i is represented by a prototype vector, $\mathbf{m}_i = [m_{i1}, \ldots, m_{id}]$ where d is input vector dimension. The units are connected to adjacent ones by a neighborhood relation. The set of neighboring map units of the map unit i is denoted by N_i. The prototype vectors define a tessellation of the input space into a set of Voronoi

sets $V_i = \{\mathbf{x} \mid \|\mathbf{x} - \mathbf{m}_i\| < \|\mathbf{x} - \mathbf{m}_j\| \, \forall j \neq i\}$. In effect each data vector belongs to the Voronoi set of the prototype to which it is closest.

Using SOM for clustering is a two-level approach: first the data is partitioned into a few hundred Voronoi sets, each corresponding to one map unit, and then the map units are clustered. All data vectors in a Voronoi set belong to the same cluster as the corresponding map unit. By clustering the SOM rather than the data directly, significant gains in the speed of clustering can be obtained [8].

Several approaches for the second level — clustering the map units — have been suggested[1]: for example a second-level SOM, k-means or some agglomerative clustering scheme [4, 5, 8]. Recently, Vellido *et al.* proposed a clustering algorithm based on distance matrices [7]. Distance matrices are visualization techniques based on showing the distances between neighboring map units [3, 6]. Since the point density of the map prototypes roughly follows the probability density function of the data, these distances are approximately inversely proportional to the density of the data, and thus distance matrices are a mode-seeking approach to clustering.

In Vellido's approach, the distance matrix is used to identify cluster centers from the SOM. The rest of the map units are then assigned to the cluster whose center is closest. This algorithm is simple and fast, but it also makes the implicit assumption that the border between two clusters lies on the middle-point between their cluster centers. In this paper, an enhanced version based on region-growing is used, which divides the map into a set of base clusters:

1. Local minima of the distance matrix are found. This is done by finding the set of map units i for which: $f(\mathbf{m}_i, N_i) \leq f(\mathbf{m}_j, N_j)$, $\forall j \in N_i$, where $f(\mathbf{m}_i, N_i)$ is some function of the set of neighborhood distances $\{\|\mathbf{m}_i - \mathbf{m}_j\| \mid j \in N_i\}$, associated with map unit i. In the experiments, median distance was used. The set of local minima may have units which are neighbors of each other. Only one minimum from each such group is retained.
2. Initialization. Let each local minimum be one cluster: $C_i = \{\mathbf{m}_i\}$. All other map units j are left unassigned.
3. Calculate distance $d(C_i, \{\mathbf{m}_j\})$ from each cluster C_i to (the cluster formed by) each unassigned map unit j.
4. Find the unassigned map unit with smallest distance and assign it to the corresponding cluster. Two optional constraints can be used to limit the growth of the clusters:
   - The continuity constraint: only those unassigned map units are considered for merging which are neighbors of the units in the clusters [5]. This ensures that the clusters form continuous areas on the map.
   - Cluster border constraint: map units on borders between clusters may have been identified beforehand using, for example, presence of empty

[1] The simplest approach is to handle the map units themselves as the final clusters. However, this is not very sensible. Because of the neighborhood relations, neighboring map units reflect the properties of the *same* rather than different clusters. To remove this overlap, the neighborhood relations can be removed ($N_i = \emptyset$, $\forall i$) in which case the SOM reduces to the k-means algorithm.

map units, as proposed in [9]. Connections to such border units can be removed, thus creating barriers for the region-growing procedure.

5. Repeat from step 3 until no more connections can be made.
6. If there are any unassigned map units, for example unconnected map units due to the cluster border constraint, they are assigned to the same cluster as the closest (neighboring) map unit.

This procedure provides a partitioning of the map into a set of base clusters, the number of which is equal to the number of local minima on the distance matrix. A problem is that the distance matrix may have some local minima which are a product of random variations in the data rather than real local maxima of the probability density function. Such base clusters should be pruned out of the clustering. In the following, this is done in a hierarchical fashion.

3 Cluster hierarchy

In cluster analysis, constructing a cluster hierarchy is often beneficial. Apart from the need for pruning above, a cluster hierarchy may represent the true structure of the data set better than a single-level partitioning: some clusters can be considered super-clusters, consisting of several sub-clusters. In data mining this allows the data to be investigated at several levels of detail.

Agglomerative clustering algorithms can be used to construct the cluster hierarchy starting from any set of base clusters (Fig. 1c-d). However, most agglomerative algorithms produce binary trees which may not be representative of the true structure. If in reality, a super-cluster consists of three (or more) sub-clusters, the binary tree will have one (or more) extra intermediate clusters which should be pruned out, for example as follows:

1. Start from root (top level) cluster.
2. For the cluster c under investigation, generate different sub-cluster sets. A sub-cluster set may contain either sub-clusters of cluster c or sub-clusters of c's sub-clusters (sub-sub-clusters).
3. Each sub-cluster set defines a partitioning of the data in the investigated cluster. Investigate each partitioning using some clustering validity measure, for example Davies-Bouldin index [1]. However, the Davies-Bouldin index depends mainly on the cluster centroids, and thus is not sensitive to local density of the data. In its stead, a measure of the gap between the two clusters can be used:

$$I_{gap} = \frac{1}{C}\sum_{i=1}^{C}\max_j\{\frac{S_i + S_j}{d_{ij}}\} \tag{1}$$

$$S_i = E\{\|\mathbf{m}_k - \mathbf{m}_l\| \mid \mathbf{m}_k, \mathbf{m}_l \in C_i, k \in N_l, V_k, V_l \neq \emptyset\} \tag{2}$$

$$d_{ij} = E\{a\|\mathbf{m}_k - \mathbf{m}_l\| \mid \mathbf{m}_k \in C_i, \mathbf{m}_l \in C_j, k \in N_l\}. \tag{3}$$

where C is the number of clusters, E is the average and $a = 2$ iff $V_k = \emptyset \vee V_l = \emptyset$, and $a = 1$ otherwise. The coefficient a is used to reflect the

fact that an empty map unit between two clusters does not really belong to either, and thus the distance between the clusters is approximately twice the distance from either cluster to the empty map unit.
4. Select the best sub-cluster set (for example the one with minimum I_{gap}), and prune the corresponding intermediate clusters.
5. Select an uninvestigated cluster (if any), and continue from step 2.

This procedure gives a pruned cluster tree (Fig. 1e) together with goodness measures of the clustering quality of the sub-cluster sets of each node in the tree. A particular partitioning is obtained from this tree by starting from the top with all data in a single cluster, and traversing the tree downwards by always splitting the intermediate node with best clustering validity index, until a predetermined number of clusters has been obtained. Alternatively, some validity index can be used to select the best number of clusters.

4 Comparisons

In this section, three different ways to cluster the SOM are compared to each other in a 2-dimensional clustering problem. The data set consists of 2200 data points divided to 7 clusters, plus 20 outlier points (Fig. 1a). The data was normalized to have unit variance in each dimension, and then a SOM with approximately 200 map units was trained[2] using the batch training algorithm and final neighborhood width of 1. After training the SOM (see Fig. 1b), the following three algorithms were used to cluster the map units from 2 upto 20 clusters[3]:

1. k-means clustering of the map prototypes
2. forming base clusters by clustering the map prototypes with region-growing, followed by agglomerative clustering of the base clusters, pruning the tree using I_{gap} and selecting the final partition as discussed above (cluster distances are calculated as distances between their centroids)
3. as previous item, but using Vellido's algorithm to form the base clusters and Davies-Bouldin index in the pruning phase

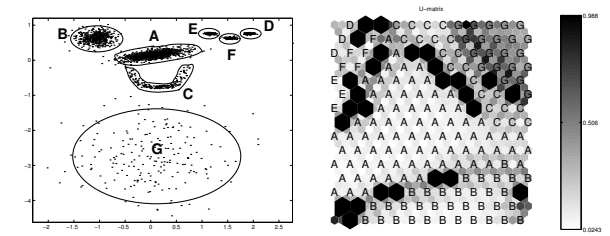
Fig. 1f-h show examples of the clustering results. These were evaluated using mutual information as a similarity measure between the true clusters $\{C_i\}$ and the acquired clusters $\{\hat{C}_i\}$:

$$m(C, \hat{C}) = -\sum_i p_i \log(p_i) - \sum_j \hat{p}_j \log(\hat{p}_j) + \sum_{i,j} p(i,j) \log(p(i,j)) \qquad (4)$$

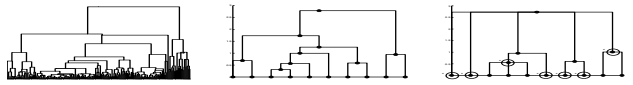
where p_i is the probability of $\mathbf{x} \in C_i$, $\hat{p}_j$ is the probability of $\mathbf{x} \in \hat{C}_j$ and $p(i,j)$ is the probability of $\mathbf{x} \in C_i \wedge \mathbf{x} \in \hat{C}_j$. The tests were repeated 20 times such that each time the data set was regenerated. The averages and standard deviations

[2] Using SOM Toolbox: `http://www.cis.hut.fi/projects/somtoolbox`

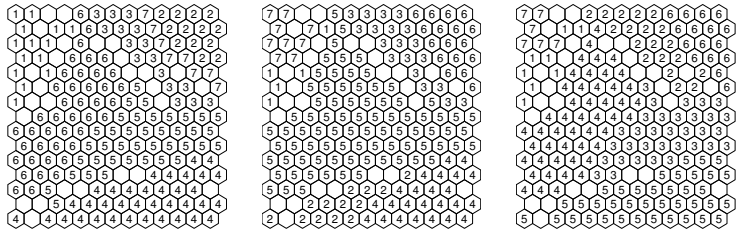
[3] Notice that the two latter algorithms are limited by the number local minima in the distance matrix, and therefore did not reach the full maxima of 20 clusters.



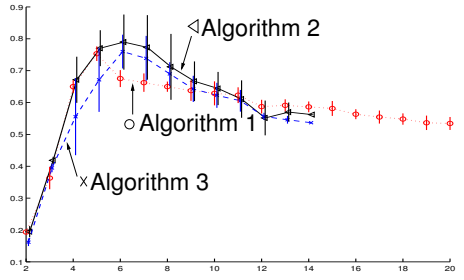
(a) True (b) U-matrix



(c) Full (d) Base (e) Pruned



(f) Algorithm 1 (g) Algorithm 2 (h) Algorithm 3



(i) Mutual information $m(C, \hat{C})$

Fig. 1. (a) Data set with true clusters indicated with encircled areas and the letters. (b) U-matrix of the data with empty map units shown as black, and correct clustering result (obtained by majority voting in each map unit) with the letters. (c) Dendrogram starting from each map unit. (d) Dendrogram starting from base clusters. (e) Pruned dendrogram, encircled clusters are those selected for (g). (f-h) Results of tested algorithms for 7 clusters, respectively. (i) Average mutual information of true and acquired clusters, and associated standard deviations (vertical lines).

of $m(C, \hat{C})$ for different algorithms and different numbers of clusters are shown in Fig. 1i.

It can be seen that the best results are produced by algorithm 2. Algorithm 1 gives best results for high ($\geq$ 12) number of clusters, but especially around 7, which is the correct number of clusters, both other algorithms give better results. Vellido's original algorithm corresponds to the third algorithm without pruning. It can be seen that the pruning produces considerable benefits.

5 Conclusion

In this paper, automated clustering of SOM based on distances between neighboring map units has been considered. The algorithm proposed by Vellido has been enhanced with additional constraints, and with an hierarchical pruning phase in order to get rid of extra clusters. The proposed algorithm has been compared with Vellido's algorithm and k-means clustering. In the tests k-means works well with small number of clusters, but its performance decreases as the number of clusters increases. Vellido's algorithm, on the other hand, produces quite too many clusters. The proposed algorithm combines Vellido's algorithm with a local density based pruning procedure and produced the best results in our experiments.

References

[1] David L. Davies and Donald W. Bouldin. A Cluster Separation Measure. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, PAMI-1(2):224–227, April 1979.

[2] Teuvo Kohonen. *Self-Organizing Maps*, volume 30 of *Springer Series in Information Sciences*. Springer, Berlin, Heidelberg, 2nd edition, 1995.

[3] Martin A. Kraaijveld, Jianchang Mao, and Anil K. Jain. A Nonlinear Projection Method Based on Kohonen's Topology Preserving Maps. *IEEE Trans. on Neural Networks*, 6(3):548–59, 1995.

[4] Jouko Lampinen and Erkki Oja. Clustering Properties of Hierarchical Self-Organizing Maps. *Journal of Mathematical Imaging and Vision*, 2(2-3):261–272, November 1992.

[5] F. Murtagh. Interpreting the Kohonen self-organizing map using contiguity-constrained clustering. *Pattern Recognition Letters*, 16:399–408, 1995.

[6] A. Ultsch and H. P. Siemon. Kohonen's Self Organizing Feature Maps for Exploratory Data Analysis. In *Proceedings of International Neural Network Conference (INNC'90)*, pages 305–308, Dordrecht, Netherlands, 1990. Kluwer.

[7] A. Vellido, P.J.G Lisboa, and K. Meehan. Segmentation of the on-line shopping market using neural networks. *Expert Systems with Applications*, 17:303–314, 1999.

[8] Juha Vesanto and Esa Alhoniemi. Clustering of the Self-Organizing Map. *IEEE Transactions on Neural Networks*, 11(2):586–600, March 2000.

[9] Xuegong Zhang and Yanda Li. Self-Organizing Map as a New Method for Clustering and Data Analysis. In *Proceedings of International Joint Conference on Neural Networks (IJCNN'93)*, pages 2448–2451, 1993.