Automatic Clustering with Self-Organizing Maps and Genetic Algorithms II: an Improved Approach

ANGEL FERNANDO KURI-MORALES
Departamento de Computación
Instituto Tecnológico Autónomo de México
Río Hondo No. 1
México 01000, D.F.
MÉXICO

Abstract. The analysis of data sets of unknown characteristics usually demands that subsets (or clusters) of the data are identified in such a way that the members of any one such cluster display common (in some sense) characteristics. In order to do this we must determine a) The number of clusters, b) The clusters themselves and c) The labeling of every element in the data set such that each element belongs uniquely to one of the clusters. In a previous work [1] we discussed an algorithm which allowed us to solve (b) and (c) (assuming that (a) is given). Further, we also showed that the so-called labeling problem may be solved by minimizing an adequate measure of distance. The metrics discussed relied on a homogeneous distribution of the samples. In this paper we discuss several metrics as applied to self-organizing maps (SOMs) which make the said consideration unnecessary and, therefore, generalize our past method. Furthermore, the new metrics improve on our previous results. We also discuss the minimization (genetic) algorithm (GA) and offer some results derived from its application.

Keywords. Self-organizing Maps, data compression, genetic algorithms.

1 Introduction

There are several ways to attempt the identification of clusters in a set of data [2], [3], [4]. If we have information regarding the source of the data we may apply classical and/or heuristic methods with relative success [5]. Here, however, we assume that nothing is known about the data under study and apply the method originally proposed by Kohonen [6] which originates the so-called self-organizing maps (SOM). In this method, basically, a set of vectors (or "neurons") η is defined. The cardinality of η ($|\eta|$) is typically smaller than that of the objects in δ (the data set) i.e. $|\eta| \leq |\delta|$. The dimensionality of every vector in δ is determined by the number of features (φ) of each object and every such object is, thus, defined in a φ-dimensional space. The neurons in a self-organizing map are simultaneously defined on two spaces: a) A φ-dimensional space of features and b) A "geographic" map of γ dimensions (typically $\gamma = 2$ or $\gamma = 3$). The training algorithm then operates on the neurons in a way such that neighboring neurons in γ space (hence the name "SOM") correspond to elements which share some (possibly non-linear) attributes in δ space. Throughout this process it is relatively simple to overcome a priori limitations present in methods which rely on classical measures of distance (such as Euclidean or Mahalanobis' [7]). Once a set of neurons is trained (i.e. once its coordinates in space are determined), however, one is faced with the problem of finding the boundaries between the neurons in a SOM which distinguish one cluster from another. A simple example will illustrate this fact. Assume that $|\delta|$=200, $|\varphi|$=4, η=16, γ=2 and that the (known) number of classes (C) is 3. Let us further assume that through some yet unspecified method the neurons corresponding to each of these 3 classes has been found, yielding a map as in Figure 1. Notice that, by definition, all neurons for C = i (i = 1, 2, 3) are "physically" close in a euclidean sense. The process of assigning a label to every group of clustered neurons usually consists of setting a class number for every neuron from a previously known classification. In our example the data would assume a form analogous to the one shown in Table 1. In this table the heading "F1" to "F4" denote features 1 to 4; class 1 consists of *k-1* elements, class 2 of *m-k* and class 3 of *200-m*

elements, respectively. There are alternative ways of displaying class membership but we will adhere to the one illustrated. When the C_i columns are known, other unknown elements which stem from the same source (H) giving rise to δ may be classified successfully, as has been shown in the past [8].

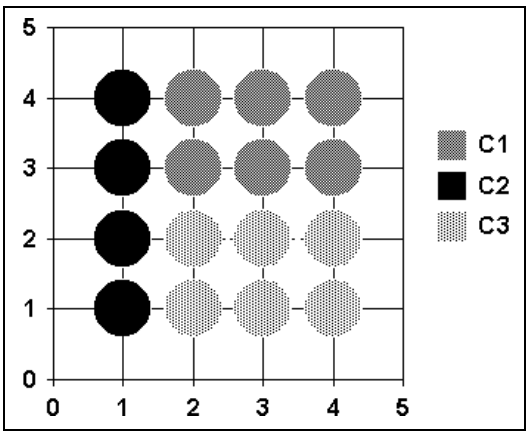

Fig. 1. A Labeled SOM with 3 classes

In the absence of the C_i columns shown in Table 1, one must resort to some method which assigns class membership to every element in δ, so as to achieve a similar tabular structure. In the example above there are 3^{200} (roughly equivalent to 10^{95}) possible assignments. The problem that we would like to solve may be simply stated as follows: Given a table such as Table 1 but lacking the data corresponding to the class columns, what is the best way to fill in these columns such that adequate clustering is achieved? We may encode any assignment as a string **S** of size $|\delta|$ consisting of a set of numbers between 1 and C. For example, if C=3 and $|\delta|$=20, the string $\mathbf{S_1}$=12312312312312332132 is one of the possible solutions; the string $\mathbf{S_2}$=22211133333322211123 is another, and so on. What we would like to answer is: which of the two possible solutions in our example is better?

Table 1. Labeling Data

F1	F2	F3	F4	C_1	C_2	C_3	
φ_{11}	φ_{12}	φ_{13}	φ_{14}	1	0	0	1
φ_{21}	φ_{22}	φ_{23}	φ_{24}	1	0	0	
...	...	...	...	1	...	...	k-1
φ_{k1}	φ_{k2}	φ_{k3}	φ_{k4}	0	1	0	k
$\varphi_{k+1,1}$	$\varphi_{k+1,2}$	$\varphi_{k+1,3}$	$\varphi_{k+1,4}$	0	1	0	
...	...	...	...	...	1	...	m-1
φ_{m1}	φ_{m2}	φ_{m3}	φ_{m4}	0	0	1	m
$\varphi_{m+1,1}$	$\varphi_{m+1,2}$	$\varphi_{m+1,3}$	$\varphi_{m+1,4}$	0	0	1	
...	...	...	...	...	...	1	200

In order to do this we must define a measure of goodness; a metric by which to judge the hypothetical solutions. The first step is to calculate a distance matrix Δ, thus:

$$\Delta_{ij} = \sum_{l=1}^{|C(l)|} \left[1/\sqrt{\sum_{k=1}^{\varphi} \left(w_{ik} - o_{ik} \right)^2} \right] \quad \begin{matrix} j = 1,...,C \\ i = 1,...,|\eta| \end{matrix} \tag{1}$$

Clearly, any exhaustive enumerative approach is out of the question. We will, therefore, appeal to a genetic algorithm (GA) to find an approximation to the best possible assignment. The detailed description of the algorithm we used may be found in [1, 10, 11]. However, before using any optimization method we must first define a measure of fitness of an assignment. One of the contributions of this paper is the identification of a family of adequate metrics.

The rest of the paper is organized as follows. In section 2 we discuss the metrics based on the *mathematical postulates*: their advantages and shortcomings. In section 3 we describe several new metrics based on the *topological postulates*: their advantages over the former. In section 4 we describe the application of the method to a set of classification problems and the results we obtained. Finally, in section 5 we offer our conclusions and point out future lines of research.

2 Mathematical Postulates

To achieve a proper fitness function we adhere to what we have called the *mathematical postulates*. They rely, basically, on the mathematical properties of the labeling matrix. To this effect we established the first of these postulates, which reads:

Postulate 1: "For every set δ it is desirable that the groups which conform it are as different as possible amongst themselves".

Following this postulate we expect that, in the labeling matrix Δ, a larger difference between the maximum and the rest of the elements of every line is a desirable property. Therefore, a criterion to distinguish "good" candidate labelings is expressed in the following metric:

$$D_4 = \sum_{i=1}^{C} \frac{1}{|C_i|} \left\{ \sum_{j=1}^{|\eta|} \left[\sum_{k=1}^{C} \sqrt{\left(\max(\Delta_{jk}) - \Delta_{jk} \right)^2} \right] \right\}_{k=1,\ldots C;\ \text{index}[\max(\Delta_{jk})] = i} \quad (2)$$

For every line in matrix Δ the element e_{max} =[max(Δ_{jk})] is selected; the euclidean distance between e_{max} and all the remaining elements of Δ_j is calculated. This distance is accumulated in a variable corresponding to the "winning" class. Once this procedure is applied to all the rows of the matrix, the norm is the summation of all the aforementioned variables where each one of the variables is divided by the rows associated to its class. If we analyze carefully D_4 we can see that it easily determines those matrices where the distance between groups is larger. However, its maximum is attained when all the *m* neurons of the map correspond to only one of the *n* groups (which is, in general, incorrect). Therefore, a further postulate was introduced:

Postulate 2: "For all δ it is desirable that all its elements are homogeneously distributed between the groups."

In this context, a homogeneous data set is one in which the cardinality of a group in the set is the same as the cardinality of any and all the remaining groups to within a specified percentage. In [1] it was determined that when the number of elements in every group was the same with 5% tolerance, the GA was able to cope with the test problems and automatically find the required clusters with an accuracy on the order of 80%. The metric where postulate 2 was included will be denoted by D_{4H}.

But homogenous data are not to be taken for granted. And considering this non-homogeneous possible case we defined new metric (which we denote by D_{4NH}) in which the postulated homogeneity is no longer considered. To do this we notice that there is a condition which may be incorporated in alternative metrics: The fact that any candidate solution should coincide with the calculated class matrix. To see why we illustrate with a simple example. Assume that γ =2, $|\eta|$ =16 and C=3. Assume, further, that Δ is as shown in table 2.

Table 1. An example of a distance matrix

i	*j*	*Distance 01*	*Distance 02*	*Distance 03*	*Class*
1	1	47.73	67.15	99.76	3
1	2	58.28	85.01	79.50	2
1	3	70.27	105.94	57.24	2
1	4	69.03	122.05	47.80	2
2	1	56.28	85.54	81.61	2
2	2	66.59	95.54	67.45	2
2	3	80.28	108.29	51.46	2
2	4	77.51	116.76	42.27	2
3	1	60.68	103.41	62.58	2
3	2	78.91	101.29	56.32	2
3	3	101.24	96.91	45.99	1
3	4	110.48	91.18	39.93	1
4	1	75.30	100.83	50.09	2
4	2	92.64	96.45	49.06	2
4	3	119.24	84.03	42.65	1
4	4	133.27	66.98	36.19	1

In Table 2, class I_m is assigned to neuron *m* from

$$I_m = \max_{n=1,\ldots,|\eta|} \{\Delta_{nm}\} \text{ for } i = 1,\ldots,|\eta| \quad (3)$$

Denoting by Σ_{Im} the class assignment derived from (3) and by Σ_{D4} the class assignment (such as the one illustrated in table 2) derived from string **S** we immediately see that $\Sigma_{D4} = f(\mathbf{S}, , \eta, \gamma)$ and that the class assignment Σ_{D4} derived from the best string $\mathbf{S}_{D4}$ is not necessarily consistent with (3). Therefore, we introduce a new metric:

$$D_{4NH} = D_{4H} - K_1\varepsilon - K_2\mu \quad (4)$$

where ε represents the number of elements in which Σ_{D4} differs from Σ_{Im}; μ is the number of classes for which no neuron is assigned and K_1 is a penalty which has to be large enough to ensure that if $\Sigma_{D4} = \Sigma_{Im}$ it will receive a much higher fitness value than the case $\Sigma_{D4} \neq \Sigma_{Im}$. By the same token, K_2 must be chosen

such that at least one neuron is assigned to each group. In D_{4NH} we abandon postulate 2 and keep D_{4H} as a tie breaking criterion when ε=0 for more than one $\mathbf{S_i}$.

3 Topological Postulates

Although working with D_{4NH} removes the need for the second mathematical postulate, the relatively modest results obtained (for which see section 5) lead us to explore a different approach, based on topological considerations regarding the SOM. These we have called the *topological postulates*. These postulates are derived from some of the intrinsic characteristics of the SOM's training algorithm.

Postulate 1. "For every SOM it is desirable that each and every one of its nodes has been labeled such that self-organization is maintained."

Postulate 2. "For each class C_i in δ it is desirable that the neurons in the trained SOM corresponding to C_i be as close(in a Euclidean sense) as possible from one another."

Postulate 3. "Given a trained SOM and $|C|$ classes, *for each class C_k in δ it is desirable that the neurons in class as far away (in a Euclidean sense) from all the remaining classes C_i (i=1,..., $|C|$, $i \neq k$)*."

In order to describe the topological metrics derived from the postulates above we now define a *kernel* which is applied in general in the metrics to be discussed:

$$K = avg\left[\sum_{\substack{i=1 \\ i<k}}^{|\eta|} \left\| \vec{x}_i - \vec{x}_k \right\|\right] \tag{5}$$

Where $\|\bullet\|$ represents the Euclidean distance and $\vec{x}_i$ is the vector associated to the i-th neuron in the SOM. Notice that the distance may be calculated in γ-space or in φ-space, and that distance may be defined for neurons which belong to the same cluster (internal or τ distance) or to different clusters (external or ξ distance). The average defined in (5) corresponds to internal or external neurons depending on whether we wish to measure relationships of neurons within one cluster or, on the contrary, if we wish to measure the distance between clusters.

For the purposes of the following discussion, we will use two subindices associated to the kernel. The first subindex will be either ξ or τ denoting external and internal distances, respectively. The second subindex, on the other hand, will be either φ or γ depending on whether the distance is calculated in the space of features or, on the contrary, on the Cartesian space of the neurons. For instance, by $K_{\tau\varphi}$ we denote the distance of a set of neurons *within* a class as measured in the space of features; likewise, by $K_{\xi\varphi}$ we denote the distance *between* classes of neurons. With this nomenclature we are in the position to define 4 new metrics which we denote as D_{5T}, D_{6T}, D_{7T} and D_{8T}, as follows:

$$D_{5T} \prec \frac{K_{\xi\varphi}}{K_{\tau\varphi} K_{\tau\gamma}} \tag{6}$$

$$D_{6T} \prec \frac{\exp\left(K_{\xi\varphi}\right)}{K_{\tau\varphi} K_{\tau\gamma}} \tag{7}$$

$$D_{7T} \prec \frac{1}{K_{\tau\varphi} K_{\tau\gamma}} \tag{8}$$

$$D_{8T} \prec \frac{1}{K_{\tau\varphi}} \tag{9}$$

In (6)-(9), the symbol "$\prec$" means *"Evaluate the denominator of the expression to the right of "$\prec$". If it is not zero, assign the value of the quotient to the metric; otherwise, assign a value zero to it"*.

We stress the fact that these metrics are the ones which stood out after trying out some of the many combinations amenable to testing. As the reader can see, here we are trying to find the best ways in which the topological characteristics stemming from Kohonen's strategy will lead us to determine the best way in which the representative neurons tend to accommodate themselves after training. In what follows we give a brief account of the results we obtained.

4 Experiments

We selected sets of data whose characteristics were known in advance, i.e. we knew a priori the number of clusters into which the information was classified and

which objects belonged to each of the clusters. We denote these 4 sets by δ_1 - δ_4. We then ran our algorithms (training and GA). Finally, we compared the known clusters and memberships with the ones the GA found.

4.1 Data Sets

Data set δ_1 was obtained from the clinical cases reported by Dr. William H. Wolberg from the Hospital of the University of Wisconsin, Madison, USA. These data was hosted by the University of California at Irvine [9]. Every element of the sample consists of 9 cytological characteristics from tumor breast tissue. These were classified in two groups labeled "Benign" and "Malign". Total number of cases: 683; 65% (444) were benign while the remaining 35% (239 cases) were malign.

Data set δ_2 was obtained from a data base consisting of 7 kinds of different outdoors pictures. These images were manually segmented in order to create a classification for each pixel. Every element in the sample represents a 3 X 3 frame. The sample was originated by the Vision Group at the University of Massachusetts. As in the previous case these data was hosted by the University of California at Irvine [9]. Total number of elements: 14; number of classes: 7. The distribution was uniform with 30 elements per group.

Data set δ_3 was obtained as follows:

a) Generate a random value between 1 and 3 and assign it to variable C.

b) Generate a random value between 0 and 1 for each of the variables u, v, w, x, y and z.

c) If C=1: $u \leftarrow \sin(u)$; $v \leftarrow \cos(v)$; $w \leftarrow \tan(w)$; $x \leftarrow \sinh(x)$; $y \leftarrow \cosh(y)$; $z \leftarrow \tanh(z)$; $F \leftarrow w^2+2x^3-3.5y^4-wxyz+2xyz^2+3.1416uv^2w^3x^4y^5z^6-.25$; assign to class 1.

d) If C=2: $u \leftarrow \cos(u)$; $v \leftarrow \tan(v)$; $w \leftarrow \sinh(w)$; $x \leftarrow \cosh(x)$; $y \leftarrow \tanh(y)$; $z \leftarrow \sin(z)$; $F \leftarrow 3.1416u^6v^5-2.71828w^4x^3+.5778y^2z$; assign to class 2.

e) If C=3: $u \leftarrow \tan(u)$; $v \leftarrow \sinh(v)$; $w \leftarrow \cosh(w)$; $x \leftarrow \tanh(x)$; $y \leftarrow \sin(y)$; $z \leftarrow \cos(z)$; $F \leftarrow w^2+2x^3-3.5-wz+2xyz^2+3.1416u^2v^2-2.71828w^2x^2+.5778y^2z^2$; assign to class 3.

Repeat steps (a) – (e) 437 times.

Distribution was as follows. Class 1: 154 elements (35.2%); class 2: 147 elements (33.5%); class 3: 136 elements (31.2%).

Data set δ_4 was obtained as follows:

Steps a) and b) as in δ_3.

c) If C=1: $u \leftarrow \sin(u)$; $v \leftarrow \cos(v)$; $w \leftarrow \tan(w)$; $x \leftarrow \sinh(x)$; $y \leftarrow \cosh(y)$; $z \leftarrow \tanh(z)$; $F \leftarrow w^2+2x^3-3.5y^4-wxyz+2xyz^2+3.1416uv^2w^3x^4y^5z^6-.25$; assign to class 1.

d) If C=2: $u \leftarrow \cos(u)$; $v \leftarrow \tan(v)$; $w \leftarrow \sinh(w)$; $x \leftarrow \cosh(x)$; $y \leftarrow \tanh(y)$; $z \leftarrow \sin(z)$; $F \leftarrow 3.1416u^6v^5-2.71828w^4x^3+.5778y^2z$; assign to class 2.

e) If C=3: $u \leftarrow \tan(u)$; $v \leftarrow \sinh(v)$; $w \leftarrow \cosh(w)$; $x \leftarrow \tanh(x)$; $y \leftarrow \sin(y)$; $z \leftarrow \cos(z)$; $F \leftarrow w^2+2x^3-wz+2xyz^2+3.1416u^2v^2-2.71828w^2x^2+.5778y^2z^2-3.5$; assign to class 3.

f) If C=4: $u \leftarrow \sinh(u)$; $v \leftarrow \cosh(v)$; $w \leftarrow \tanh(w)$; $x \leftarrow \sin(x)$; $y \leftarrow \cos(y)$; $z \leftarrow \tan(z)$; $F \leftarrow 3.1416u^6v^5+w^2+2x^3+2xyz^2+.5778y^2z^2+uwy-vxz$; assign to class 4.

Repeat steps (a) – (f) 437 times.

Distribution was as follows. Class 1: 109 elements (24.9%); class 2: 122 elements (27.9%); class 3: 107 elements (24.5%); class 4: 99 elements (22.7%).

The GA operating with D_{4H} and D_{4NH} performed as shown in table 3. In the table, the unlabeled row refers to the results obtained by training a SOM with the traditional supervised labeling. This tables reflects the best values we were able to obtain from metrics which only considered the mathematical properties of the classical labeling algorithm. Likewise, in table 4 we show the results of applying the metrics derived from topological considerations.

Table 3. Performance of D4H and D4NH

	Efficiency (%)			
	δ_1	δ_2	δ_3	δ_4
D_{4H}	83.54	56.37	82.58	72.00
D_{4NH}	94.73	57.14	79.60	89.5
	94.73	79.50	97.50	96.10

Table 4. Performance of D_{5T}, D_{6T}, D_{7T} and D_{8T}

	Efficiency (%)			
	δ_1	δ_2	δ_3	δ_4
D_{5T}	70.14	61.42	97.48	76.88
D_{6T}	70.13	57.61	97.48	78.95
D_{7T}	95.31	66.19	97.48	64.30
D_{8T}	94.73	58.09	97.48	78.95
	94.73	79.50	97.50	96.10

5 Conclusions

From tables 3 and 4, two conclusions are immediate:

a) The metrics defined from the mathematical postulates are weaker than those arising from the topological postulates.
b) Topologic metrics denoted as D_{7T} and D_{8T} yield better results than the other ones, in general.

The problem of automatic clustering is a very complex one and, given the exceedingly large search spaces, the previous results seem to be impressive. It is clear that no absolute conclusions may be inferred from our analysis which, up to this point, has been qualitative. To establish hard conclusions we have to analyze a very large number of problems of this sort. A statistical methodology has been already developed [12] with good results. We intend to proceed our investigation resorting to this methodology. Until then, we have no way to ascertain the "hardness" of our results.

However, an even summary review of the literature having to do with the problem we are tackling clearly leaves no doubt as to the interest of the results reported here because:

a) We are assuming no knowledge of the data with which we are working.
b) We are not assuming a preconceived form of the relationships involved in the clustering.
c) We do not rely on an expert to establish some kind of rule in order to determine the relationship we are looking for in the elements of a cluster.
d) We are discarding the need to have prior knowledge of the clusters involved in order to be able to label the clusters so as to make generalization possible.
e) We are dealing with an NP kind of problem (an issue we are not stopping to prove) and done so with success via a Genetic Algorithm.
f) Other alternative methods only cope with conditions such as the one stated above in a very restricted sense and, usually, demand the participation of a human expert.

We believe that all these facts regarding the method reported here should be sufficient to point at the practicality of these strategy, particularly in the field of knowledge discovery and data mining.

We expect to report soon enough on hard (statistically supported) results which confirm our results to date.

References

[1] Kuri, A., "Automatic Clustering with Self-Organizing Maps and Genetic Algorithms", Recent Advances in Simulation, Computational Methods and Soft Computing, pp. 173-180, WSEAS Press, 2002.

[2] Devijver, P.A. and Kittler, J., Pattern Recognition: A Statistical Approach, Prentice-Hall International, Englewood Cliffs, NJ, 1980.

[3] Duda, R. O. and Hart, P. E., Pattern Classification and Scene Analysis, Wiley-Interscience, New York, 1973.

[4] Fukunaga, K., Introduction to Statistical Pattern Recognition, 2nd Ed., Academic Press, New York, 1990.

[5] Schalkoff, R., Pattern Recognition: Statistical, Structural and Neural Approaches, John Wiley & Sons, New York, 1992.

[6] Kohonen, T., Self-Organizing Maps, Springer, Berlin, 2001.

[7] Duda, D., and Hart, P.E., op. cit.

[8] Kohonen, T., Barna, G. and Chrisley, R., Statistical Pattern Recognition with neural networks. Benchmarking studies, IEEE International Conference on Neural Networks, vol I, pp. 61-68, 1988, San Diego, CA.

[9] www.ics.uci.edu/~mlearn/MLRepository.html

[10] Kuri, A., "A Universal Eclectic Genetic Algorithm for Constrained Optimization", 1998, Proceedings 6th European Congress on Intelligent Techniques & Soft Computing, EUFIT'98, pp. 518-522.

[11] Kuri, A., A Comprehensive Approach to Genetic Algorithms in Optimization and Learning. Theory and Applications, Vol. 1. Instituto Politécnico Nacional, pp 270, 1999.

[12] Kuri, A., A Methodology for the Statistical Characterization of Genetic Algorithms, Proceedings of the II Mexican International Congress on Artificial Intelligence, MICAI 2002, LNAI 2313, pp. 79-88.