computing@computingonline.net
www.computingonline.net

ISSN 1727-6209
International Journal of Computing

SIMULTANEOUS TOPOLOGICAL CATEGORICAL DATA CLUSTERING AND CLUSTER CHARACTERIZATION

Lazhar Labiod, Nistor Grozavu, Younès Bennani

LIPN – UMR CNRS 7030 Institut Galilée – University Paris-Nord
99, avenue Jean-Baptiste Clément 93430 Villetaneuse, France
{lazhar.labiod, nistor.grozavu, younes.bennani}@lipn.univ-paris13.fr

Abstract: *In this paper we propose a new automatic learning model which allows the simultaneously topological clustering and feature selection for quantitative datasets. We explore a new topological organization algorithm for categorical data clustering and visualization named RTC (Relational Topological Clustering). Generally, it is more difficult to perform clustering on categorical data than on numerical data due to the absence of the ordered property in the data. The proposed approach is based on the self-organization principle of the Kohonen's model and uses the Relational Analysis formalism by optimizing a cost function defined as a modified Condorcet criterion. We propose an iterative algorithm, which deals linearly with large datasets, provides a natural clusters identification and allows a visualization of the clustering result on a two dimensional grid. Thereafter, the statistical ScreeTest is used to detect relevant and correlated features (or modalities) for each prototype. This test allows to detect the most important variables in an automatic way without setting any parameters. The proposed approach was validated on variant real datasets and the experimental results show the effectiveness of the proposed procedure.*

Keywords: *Topological learning, Relational Analysis, Categorical data, Features selection.*

1. INTRODUCTION

In the exploratory data analysis of high dimensional data, one of the main tasks is the formation of a simplified, usually visual, overview of datasets. This can be achieved through simplified description or summaries, which should provide the possibility to discover most relevant features or patterns. Clustering and projection are among the examples of useful methods to achieve this task. On one hand classical clustering algorithms produce groupes of data according to a chosen criterion. Projection methods, on the other hand, represent the data points in a lower dimensional space in such a way that the clusters and the metric relations of the data items are preserved as faithfully as possible. In this field, most algorithms use similarity measures based on Euclidean distance. However there are several types of data where the use of this measure is not adequate. This is the case when using categorical data since, generally, there is no known ordering between the feature values. In this work, we present a new formalism that can be applied to this type of data and simultaneously achieves the both tasks, clustering and visualization.

Topological learning is a recent direction in Machine Learning which aims to develop methods grounded on statistics to recover the topological invariants from the observed data points. Most of the existed topological learning approaches are based on graph theory or graph-based clustering methods.

The topological learning is one of the most known technique which allow clustering and visualization simultaneously. At the end of the topographic learning, the "similar" data will be collect in clusters, which correspond to the sets of similar observations. These clusters can be represented by more concise information than the brutal listing of their patterns, such as their gravity center or different statistical moments. As expected, this information is easier to manipulate than the original data points. The neural networks based techniques are the most adapted to topological learning as these approaches represent already a network (graph). This is why, we use the principle of the self-organizing maps which represent a two layer neural network: an entry layer (the data) and a topological layer (the map).

In order to visualize the partition obtained by the Relational Analysis approach (Marcotorchino, 2006) [12], (Marcotorchino and Michaud, 1978) [13] the authors proposed a methodology called "Relational Factorial Analysis" (Marcotorchino, 1991, 2000) [14; 15] which combines the Relational Analysis for

clustering and the Factorial Analysis for the visualization of the partition on the factorial designs. It is a juxtaposition of the both methods, the methodology presented here combines the Relational Analysis approach and the SOM principle determined by a specific formalism to this methodology. The proposed model allows simultaneously, to achieve data clustering and visualization. It automatically provides a natural partition of the data (i.e without fixing a priori the number of clusters and the size of each cluster) and a self-organization of the clusters on a two-dimensional map while preserving the a priori topological data structure (i.e two close clusters on the map consist of close observations in the input space). Various methods based on the principle of the SOM model were proposed in the literature for binary data processing: probabilistic methods and others quantization techniques. Most of these methods operate on the data after a preliminary transformation step in order to find a continuous representation of the data, and then apply the SOM model, as KACM (Cottrell and Letremy, 2003) [3] and the approach suggested by Leich and al (Leich, Weingessel and Dimitriadou, 1998) [10]. These methods destroy the binary nature of the data, in other words, they violate the structure of the data to meet the requirements of the method. In (Lebbah, Badran and Thiria, 2000) [7] the authors propose BTM (Binary Topological Map) method which operates directly on binary data based on the Hamming distance. In (Lebbah, Rogovschi and Bennani, 2007) [8] a probabilistic version of the SOM model is proposed, based on the Bernoulli distribution adapted to the binary data (BeSOM). The BeSOM method is an interesting approach because it allows to build a self-organizing map learned from categorical data. This is why we use this method to compare it with the proposed RTC approach. The disadvantage of these methods is that they increase the complexity compared to classical topological clustering algorithms (as SOM).

With the advent of high throughput technologies, dimensionality reduction has become increasingly important in data mining field. Its goal is to reduce the number of observations (samples) and to extract the most relevant information for each data [5]. Machine learning has been very successful in developing supervised and unsupervised learning algorithms for a wide range of technical applications where clustering and unsupervised feature selection are one of the most difficult tasks of this domain.

Clustering categorical data and categorical feature selection, i.e. data in which attribute values are not ordered, is a fundamental problem in data analysis. Moreover, most algorithms for clustering categorical data require the careful choice of parameter values, which makes these algorithms difficult to be used by a non-expert with the method.

In literature were proposed an extension of the SOM model to detect the relevant features by introducing a weighting technique called lwo-SOM [17]. Continuous weighting provides more information about the relevance of various features, and topological clustering and feature weighting are thus clearly linked. In contrast to SOM and lwo-SOM approaches, which deals with continuous data and has several parameters to set (the map size, the learning rate, weight vectors), the Relational Topological Clustering [19] allows to cluster categorical data without setting any parameter (the only parameter is for the visualization). The use of the RTC method will be the first step of our model across the automatic adaptive learning.

Feature selection is commonly used in machine learning, wherein a subset of the features available from the data are selected for application of a learning algorithm. The best subset contains the features that give the highest accuracy score. This is an important stage of preprocessing and is one of two ways of avoiding the curse of dimensionality.

The number of observations can be reduced through unsupervised learning and feature selection. The importance of each feature depends on the size of the learning dataset – for a small sample size, eliminating a relevant feature can reduce the error. Note also that irrelevant features can be very informative when used together.

In this paper, we consider the both cases: to reduce the data size and to eliminate the noisy features from this data. To reduce the number of observations we use the proposed RTC method to build a prototype matrix which will represent the dataset. Thereafter, for variable selection task we use the statistical approach Scree Test of Cattell which is initially proposed to select the principal components [28].

This paper is organized in the following way: in section 2 we present the Relational Analysis approach for clustering, section 3 presents the topological clustering and features selection problems. Section 4 shows the proposed relational topological clustering called RTC. We present the proposed automatic learning system in section 5 which allows topological clustering and feature selection simultaneously. In section 6, we show the experimental results obtained for several datasets. Some conclusions and future perspectives are discussed at the end of the paper.

2. RELATIONAL ANALYSIS APPROACH

Relational Analysis was developed in 1977 by F. Marcotorchino and P. Michaud, inspired by the work of Marquis de Condorcet, which was interested in the 18th century with the result of collective vote starting from individual votes. This methodology is based on the relational representation (pairwise comparison) of data objects and the optimization under constraints of the Condorcet criterion.

2.1. DEFINITIONS AND NOTATIONS

Let D be a dataset with a set I of N objects $(O_1, O_2, ..., O_N)$ described by the set V of M categorical attributes (or variables) $V^1, V^2., V^m, .., V^M$, each one having $p_1, .., p_m, .., p_M$ categories respectively and let $P = \sum_{m=1}^{M} p_m$ to denote the full number of categories of all variables. Each categorical variable can be decomposed into a collection of indicator variables. For each variable V^m, let the p_m values to correspond to the numbers from 1 to p_m and let $V_1^m, V_2^m, ..., V_{p_m}^m$ be the binary variables such that for each j, $1 \le j \le p_m$, $V_k^m = 1$ if and only if the V^m takes the j-th value. Then the dataset can be expressed as a collection of M matrices K^m $(N \times p_m)$ (for $m = 1, .., M$) with general term k_{ij}^m such as:

$$k_{ij}^m = \begin{cases} 1 & \text{if the object } i \text{ takes the categorie } j \text{ of } V^m \\ 0 & \text{otherwise} \end{cases} \quad (1)$$

which gives the N by P binary disjunctive matrix $K = (K^1 \mid K^2 \mid ... \mid K^m \mid ... \mid K^M)$.

2.2. RELATIONAL DATA REPRESENTATION

If a dataset is made up of N objects $(O_1, O_2, ..., O_N)$ on which M attributes (or variables) $(V^1, V^2, ..., V^M)$ have been measured then the "pairwise comparison principle" consists in transforming this dataset, which is usually, represented by a $N \times M$ rectangular matrix into two squared $N \times N$ matrices S and $\bar{S}$. The matrix S, which is called the global relational Condorcet's matrix, of general term $s_{ii'}$ represents the global similarity measure between the two objects O_i and $O_{i'}$ over all the M attributes and matrix $\bar{S}$ of general term $\bar{s}_{ii'}$ which represent the global dissimilarity measure of these two objects. To get matrix S, each V^m attribute is transformed into a squared $N \times N$ matrix S^m of general term $s_{ii'}^m$ which represent the similarity measure between the two objects O_i and $O_{i'}$ with regards to attribute V^m. Then, $s_{ii'}^m = 1$ if O_i and $O_{i'}$ take the same categorie of V^m and 0 otherwise. To get matrix $\bar{S}$, a dissimilarity measure $\bar{s}_{ii'}^m$ of objects O_i and $O_{i'}$ with regards to the attribute V^m is then computed as the complement to the maximum possible similarity measure between these two objects. As the similarity between two different objects is less or equal to their self-similarities: $s_{ii'}^m \le min(s_{ii}^m, s_{i'i'}^m)$ then $\bar{s}_{ii'}^m = \frac{1}{2}(s_{ii}^m + s_{i'i'}^m) - s_{ii'}^m$. This leads to a dissimilarity measure matrix $\bar{S}^m$. The matrices S and $\bar{S}$ are then obtained by summing, respectively, all the matrices S^m and $\bar{S}^m$, that is $S = \sum_{m=1}^{M} S^m$ and $\bar{S} = \sum_{m=1}^{M} \bar{S}^m$. The global similarity between each two objects O_i and $O_{i'}$ is thus $s_{ii'} = \sum_{m=1}^{M} s_{ii'}^m$ and their global dissimilarity is $\bar{s}_{ii'} = \sum_{m=1}^{M} \bar{s}_{ii'}^m$.

2.3. MAXIMIZATION OF THE CONDORCET'S CRITERION

To cluster a population of N objects described by M variables, the relational analysis theory maximises the Condorcet's criterion:

$$\max_X \mathsf{R}_{RA}(S, X)$$

with $X = \{x_{ii'}\}_{i,i':1,...,N}$ representing an equivalence relation defined on $I \times I$.
Where

$$\mathsf{R}_{RA}(S, X) = \sum_{i,i'=1}^{N} s_{ii'} x_{ii'} + \sum_{i,i'=1}^{N} \bar{s}_{ii'} \bar{x}_{ii'} \quad (2)$$

$$= \sum_{i,i'=1}^{N} (s_{ii'} - \bar{s}_{ii'}) x_{ii'} + \sum_{i,i'=1}^{N} \bar{s}_{ii'} \quad (3)$$

$$= 2 \sum_{i,i'=1}^{N} (s_{ii'} - \frac{1}{2} \frac{s_{ii} + s_{i'i'}}{2}) x_{ii'} + \beta \quad (4)$$

Where $\beta = \sum_{i,i'=1}^{N} \bar{s}_{ii'}$ is a constant term, and X is the reached solution wich models a partition in a relational space (an equivalence relation), and must check the following properties:

$$\begin{cases} x_{ii} = 1, \forall i & \textit{reflexivity} \\ x_{ii'} - x_{i'i} = 0, \forall (i,i') & \textit{symmetry} \\ x_{ii'} + x_{i'i''} - x_{ii''} \leq 1, \forall (i,i',i'') & \textit{transitivity} \\ x_{ii'} \in \{0,1\}, \forall (i,i') & \textit{binarity} \end{cases}$$

Let us consider $\mathsf{C} = \{\mathsf{C}_1,..\mathsf{C}_l,...\mathsf{C}_L\}$ a partition of the set I into L clusters, the Condorcet criterion breaks up into terms of contributions where the contribution $cont(i,l)$ of an object i in a cluster C_l is written:

$$cont(i,l) = \sum_{i' \in \mathsf{C}_l} [s_{ii'} - \alpha(\frac{s_{ii} + s_{i'i'}}{2})] \qquad (5)$$

Where $\alpha \in [0,1]$ is the similarity threshold, and we have

$$\mathsf{R}_{RA}(S,X) = \sum_{i=1}^{N}\sum_{l=1}^{L} cont(i,l) \qquad (6)$$

That we can express in terms of the object profile K_i representing the i^{th} row of the complete disjunctive table K and P_l the prototype of cluster C_l, is defined in the following way:

$$s_{ii'} =< K_i, K_{i'} > \ and \ \ P_l = \sum_{i' \in \mathsf{C}_l} K_{i'} \qquad (7)$$

Then, we have

$$cont(K_i, P_l) =< K_i, P_l > -\alpha S_{il} \qquad (8)$$

Where $S_{il} = \dfrac{|\mathsf{C}_l| < K_i, K_i > + \sum_{i' \in \mathsf{C}_l} < K_{i'}, K_{i'} >}{2}$.

This new formula of the contribution avoids the computation of square matrices S and $\bar{S}$ (Condorcet's matrix and its complementary) which reduces considerably the computational cost related to the contributions computation.

2.4. RELATIONAL ANALYSIS HEURISTIC

The heuristic process consists in starting from an initial cluster (a singleton cluster) and build a partition of the set I in an incremental way, by accentuating the value of Condorcet criterion $\mathsf{R}_{RA}(S,X)$ at each assignment. We give below the description of the Relational Analysis algorithm which was used by the Relational Analysis methodology (see Marcotorchino and Michaud for further details). The presented algorithm aims at maximizing the criterion given in (4) based on the contribution computation.

Algorithm1: RA heuristic
Inputs:
L_{max} = maximal number of clusters, N_{iter} = number of iterations, N = number of examples (objects), α = similarity threshold
- take the first object as the first element of the first cluster.
- $l = 1$ where l is the current number of clusters
for t=1 to N_{iter} **do**
for $i = 1$ to N **do**
for $j = 1$ to l **do**
Compute the contribution of the object i: $cont(i, j)$
end for
$l^* = arg \max_j cont(i, j)$,
where l^* is the cluster id which has the highest contribution with the object i:
$cont(i, l^*) \leftarrow$ the computed contribution
if $cont(i, l^*) < 0$ **and** $l < L_{max}$ **then**
create a new cluster where the object i is the first element;
$l \leftarrow l + 1$
else
assign object i to cluster C_{l^*}
endif
endfor
endfor
Output: at most L_{max} clusters

We have to fix a number of iterations and the similarity threshold in order to have an approximate solution in a reasonable processing time. Besides, it is also required a maximum number of clusters, but since we don't need to fix this parameter, we put by default $L_{max} = N$. Basically, this algorithm has $O(N_{iter} \times L_{max} \times N)$ computation cost. In general term, we can assume that $N_{iter} << N$, but not

$L_{max} << N$. Thus, in the worst case, the algorithm has $O(L_{max} \times N)$ computation cost.

3. TOPOLOGICAL CLUSTERING AND FEATURE SELECTION

Feature selection for clustering or unsupervised feature selection is used to identify the feature subsets that accurately describe the clusters. This improves the interpretability of the induced model, as only relevant features are involved in it, without degrading its descriptive accuracy. To produce a clustering and to visualize the clustering result, the topological clustering is the most used. This is why we will use the principle of the Self-Organizing Maps (SOM) to develop our model.

3.1. SELF-ORGANIZING MAP

The model called Kohonen's Self-Organizing Map (SOM) is an artificial neural network, which learns to model a data space $(Z, z_i \in \mathsf{R}^d)$ also called set of observations (objects) by a set of prototypes $(W, w_l \in \mathsf{R}^d)$ (the neurons) where observations and neurons are vectors of the input space.

If the network consists of L neurons, the SOM technique provides a partition into L clusters of the input space where the number of observations $N >> L$. Each neuron l is associated with a vector of weight w_l which belongs to the input space. Thus, for a set of observations the network learns the position in this space of L centers. For example in the trivial case where $L = N$, the best possible partition is obviously a discrete partition where each observation is isolated in a cluster (the center of each cluster corresponds to the observation forming the cluster), which minimizes the distance to all data objects.

The modelling quality depends on the used metric distance in a vector space. We use the Euclidean distance to measure the distance between an observation and a prototype (two vectors). In addition, to model inputs through prototypes, a self-organizing map C allows to build a graph G to structure this space and provides a visualization in one or two dimensions of the topological links between clusters. It should be remembered that the Kohonen's network is not a simple clustering algorithm, it is a model that seeks to project multidimensional observations on a discrete space (the map C) of small dimensions (usually 1, 2 or 3). This projection has to respect the property of topology "conservation" of the data, ie two neurons l, r which are neighbors over the discrete topological map must be associated with two close prototypes w_l, w_r compared to the Euclidean distance in the observation space.

The map C is in the form of an undirected graph $G = (\mathsf{C}, \mathsf{A})$, where C refers to the L vertices (neurons) and A the set of edges that gives the organization of neurons on the map C . Thus, two neurons l, r are directly connected neighbors in the map if $a(c, r) \in \mathsf{A}$. This graph induces a discrete distance δ on the map: for any pair of neurons (l, r) of the map the distance $\delta(l, r)$ is defined as being the length of the shortest path between l and r . For every neuron l , this distance determines the neighborhood of order d of c as following: $\mathsf{V}_c(d) = \{l \in \mathsf{C}, \delta(c, l) \le d\}$

This notion of neighborhood can be formalized using a kernel function $\mathbf{K}$ defined from R^+ in R^+ , and decreasing such that $\mathbf{K}(0) = 1$ and $\lim_{x \to \infty} \mathbf{K}(x) = 0$ (in practice we use $\mathbf{K}(x) = e^{-x^2}$). This function generates a family of functions $\mathbf{K}^T$, defined by $\mathbf{K}^T(x) = \mathbf{K}(\frac{x}{T})$. The parameter T is analogous to a temperature, when T is hight, then $\mathbf{K}^T(x)$ remains close to 1 even for large values of x ; contrarily a low value produces a $\mathbf{K}^T$ function which decreases quickly to 0 . The role of $\mathbf{K}^T$ is to transform the discrete distance δ induced by the structure of the graph into a regular neighborhood parameterized by T . We will use $\mathbf{K}^T_{(\delta(l,r))}$ as a measure of effective closeness between neurons l and r . During the SOM algorithm, the value of T decreases to stabilize the solution.

The quality of the partition and topology conservation is measured using the objective function $\mathsf{R}^T_{SOM}(\varphi, W)$, which should be as low as possible.

$$\mathsf{R}^T_{SOM}(\varphi, W) = \sum_{i=1}^{N} \sum_{l=1}^{L} \mathbf{K}^T_{(\delta(\varphi(i), l))} \| z_i - w_l \|^2 \quad (9)$$

Where φ represents the assignment function such that: $\varphi(i) = l \ \ if \ \ i \in \mathsf{C}_l$.

3.2. FEATURE SELECTION WITHIN SOM

Additionally, the identification of relevant and irrelevant features with SOM learning provides valuable insight into the nature of the cluster-structure.

Feature selection for clustering analysis is difficult because, unlike supervised learning, there are no class labels for the dataset and no obvious criteria to guide the search [26]. In [17], the weights Π and prototype's set W provided by lwo-SOM is used to cluster the map and to compute the relevence of the continuous features which characterize the resulting clusters associated with cells and group of cells.

The weighted SOM method were based on initial work describing the supervised model w-LVQ2 [27]. This approach adapts weights to filter the observation during the learning process. Using this model, the observations $\mathbf{x}$ were weighted using weight vectors π before computing the distance. The objective function is rewritten as follows:

$$R_{lwo}(\chi, W, \Pi) = \sum_{i=1}^{|E|}\sum_{j=1}^{|W|} K_{j,\varphi(\mathbf{x}_i)} \| \boldsymbol{\pi}_j \mathbf{x}_i - \mathbf{w}_j \|^2 \quad (10)$$

$\varphi(\mathbf{x}_i)$ is the assignment function which allows to find the Best Matching Unit (BMU), it selects the neuron with the closest prototype from the data x_i using the Euclidean distance. $K_{j,\varphi(\mathbf{x}_i)}$ is the neighborhood function on the SOM map between two cells.

Minimization of $R_{lwo}(\varphi, W, \Pi)$ was performed by iterative repetition of the following three steps until stabilization.

The initialization step determines the prototype set W and the set of associated weights Π, at each training step $(t+1)$. An observation $\mathbf{x}_i$ is then randomly chosen from the input dataset and the following operations are repeated:

• Minimize $R_{lwo}(\varphi, W, \hat{\Pi})$ with respect to φ by fixing W and Π. Each weighted observation $(\pi_j x_i)$ is assigned to the closest prototype $\mathbf{w}_j$ using the assignment function, defined as follows:

$$\varphi(\mathbf{x}_i) = \arg\min_j \left(\| \boldsymbol{\pi}_j \mathbf{x}_i - \mathbf{w}_j \|^2 \right)$$

• Minimize $R_{lwo}(\hat{\varphi}, W, \hat{\Pi})$ with respect to W by fixing φ and Π. The prototype vectors are updated using the gradient stochastic expression:

$$w_j(t+1) = w_j(t) + \varepsilon(t) K_{j,\varphi(x_i)} (\pi_j x_i - w_j(t))$$

• Minimize $R_{lwo}(\hat{\varphi}, W, \hat{\Pi})$ with respect to Π by fixing φ and W. The update rule for the feature weight vector $\pi_j(t+1)$ is:

$$\pi_j(t+1) = \pi_j(t) + \varepsilon(t) K_{j,\varphi(x_i)} x_i (\pi_j(t) x_i - w_j(t))$$

As in the traditional stochastic learning algorithm of Kohonen [5], the learning rate at time t is denoted by $\varepsilon(t)$. The training is usually performed in two phases. In the first phase, a high initial learning rate $\varepsilon(0)$ and a large neighborhood radius T_{max} are used. In the second phase, a low learning rate and small neighborhood radius are used from the beginning.

To select the relevant features associated to the most important weights, an established statistical method *scree method* were used on the computed weights. The lwo-SOM method require a low computational time but deals only for the continuous data and requires some parameters to be defined as the learning rate, the map size, the weights.

Using the principle of the cluster characterization technique combined with lwo-SOM map, in the next section we present a new procedure to cluster and to select relevant categorical features in an automatic way.

4. RELATIONAL TOPOLOGICAL CLUSTERING (RTC)

Similarly to the classical model of self-organizing map (SOM), we use for the proposed RTC model an artificial neural network with an entry layer for the observations (data) and a map C having a topological order for the exit. The topology of the map is defined via an undirected graph. Like the SOM algorithm, the RTC model includes the vector quantization procedure. During this procedure, each neuron of the map which is the index of a prototype for required quantization will be represented by a vector of the same dimension than the observations. Contrarily to SOM approach, quantization is done by means of assignment function φ adapted to binary data, the choice of prototypes and the assignment function is done by maximizing the objective function denoted $\mathsf{R}^T_{RTC}(\varphi, P)$. Maximization must allow on one hand, to define prototypes making possible to carry out a conservation of the data topology (defined by a measurement of contribution) and to carry out, on the other hand, a partition of set I into homogeneous sub sets.

The basic idea of the RTC approach is to

maximize a new objective function defined from the classical RA criterion R_{RA} by adding a regularization term R_{Topo}, which introduces a topological constraint. The RTC objective function is the follows:

$$\mathsf{R}^{T}_{RTC}(\varphi,\mathsf{X}) = \mathsf{R}_{RA}(S,X) + \mathsf{R}_{Topo}(\varphi,\mathsf{X}) \quad (11)$$

where

$$\mathsf{R}_{RA}(S,X) = \sum_{i,i'=1}^{N} \Psi_{ii'} x_{ii'} \quad (12)$$

and

$$\mathsf{R}_{Topo}(\varphi,\mathsf{X}) = \sum_{i,i'=1}^{N} \Psi_{ii'} \sum_{l=1}^{L} \mathbf{K}^{T}_{(\delta(\varphi(i),l))} \mathsf{X}_{i'l} \overline{\mathsf{X}}_{il} \quad (13)$$

Where $\forall i,i'\ \Psi_{ii'} = s_{ii'} - \alpha(\frac{s_{ii}+s_{i'i'}}{2})$, X_{il} is the general term of the partition matrix X of set I into L clusters such that $\mathsf{X}_{il} \in \{0,1\}$, $\sum_{l=1}^{L} \mathsf{X}_{il} = 1$, $\overline{\mathsf{X}}_{il} = 1 - \mathsf{X}_{il}$ and $\forall i,i';\ x_{ii'} = \sum_{l} \mathsf{X}_{il}\mathsf{X}_{i'l}$, which is the general term of the equivalence relation X.

This function breaks up into two terms, the first one corresponds to the Condorcet criterion $\mathsf{R}_{RA}(S,X)$ whose maximization makes possible to obtain a partition of I more compact possible within the meaning of the Condorcet criterion. The second term makes possible to take into account the influence of neighborhood between a neuron and its neighbors on the map C. It makes possible to bring closer the partitions corresponding to two different neurons on the map in order to preserve the topological order between the various partitions. Indeed, the second term imposes to the prototype of the neuron l to represent objects belonging to nearby neurons: if the neuron l is close to the neuron $\varphi(i)$ on the map C, a small value $[\sum_{i'=1}^{N} \Psi_{ii'} \mathbf{K}^{T}_{(\delta(\varphi(i),l))} \mathsf{X}_{i'l}]$ will more penalizes the maximization of the objective function.

The temperature T adjusts the relative importance granted to both terms. Indeed, for the large values of temperature, the second term is dominating and in this case the priority is given to the topology. More T is small, more the first term is taken into account and the priority is given to the determination of prototypes representing the compact partition. The RTC approach acts in this case exactly like the Condorcean method. It is thus possible to constat that the Relational Topological Map model makes possible to obtain a regularized solution of that obtained by the Condorcean method where the regularization is obtained by the respect of the a priori topological data structure.

The development of the both terms (12) and (13) leads to the following expression of the objective function:

$$\mathsf{R}^{T}_{RTC}(\varphi,\mathsf{X}) = \sum_{i,i'=1}^{N} \Psi_{ii'} \sum_{l=1}^{L} \mathbf{K}^{T}_{(\delta(l,l))} \mathsf{X}_{i'l}\mathsf{X}_{il} + \sum_{i,i'=1}^{N} \Psi_{ii'} \sum_{l=1}^{L} \mathbf{K}^{T}_{(\delta(\varphi(i),l))} \mathsf{X}_{i'l} \overline{\mathsf{X}}_{il}$$

$$= \sum_{i,i'=1}^{N} \Psi_{ii'} \sum_{l=1}^{L} \mathbf{K}^{T}_{(\delta(\varphi(i),l))} \mathsf{X}_{i'l} (\mathsf{X}_{il} + \overline{\mathsf{X}}_{il}) = \sum_{i,i'=1}^{N} \Psi_{ii'} \sum_{l=1}^{L} \mathbf{K}^{T}_{(\delta(\varphi(i),l))} \mathsf{X}_{i'l} \quad (14)$$

4.1. A NEW WRITING OF THE OBJECTIVE FUNCTION

The objective function above can be expressed using the profiles K_i of each object and the prototype P_l of each cell of the map C as following:

$$\mathsf{R}^{T}_{RTC}(\varphi,\mathsf{X}) = \sum_{i=1}^{N} \sum_{l=1}^{L} \mathbf{K}^{T}_{(\delta(\varphi(i),l))} \underbrace{\sum_{i'=1}^{N} \Psi_{ii'} \mathsf{X}_{i'l}}_{cont(i,l)} \quad (15)$$

Replacing the contribution $cont(i,l)$ by $cont(K_i,P_l)$ gives the following writing:

$$\mathsf{R}^{T}_{RTC}(\varphi,P) = \sum_{i=1}^{N} \sum_{l=1}^{L} \mathbf{K}^{T}_{(\delta(\varphi(i),l))} (< K_i, P_l > - \alpha S_{il}) \quad (16)$$

$$= \sum_{i=1}^{N} cont^{T}(K_i, P_{\varphi(i)}) \quad (17)$$

where

$$cont^{T}(K_i, P_{\varphi(i)}) = \sum_{l=1}^{L} \mathbf{K}^{T}_{(\delta(\varphi(i),l))} (< K_i, P_l > - \alpha S_{il}) \quad (18)$$

is the regularized contribution of the object i to his winner neuron $\varphi(i)$. We observe that the regularized contribution of the object i to $\varphi(i)$ is a weighted sum of the contributions of i to all prototypes $P_l (l = 1,...L)$ in the influence

neighborhood of $\varphi(i)$.

We can rewrite this contribution in the following simplified form:

$$cont^T(K_i, P_{\varphi(i)}) = < K_i, P_l^T > -\alpha \sum_{l=1}^{L} \mathbf{K}^T_{(\delta(\varphi(i),l))} S_{il} \quad (19)$$

where

$$P^T_{\varphi(i)} = \sum_{l=1}^{L} \mathbf{K}^T_{(\delta(\varphi(i),l))} P_l = \sum_{l=1}^{L} \mathbf{K}^T_{(\delta(\varphi(i),l))} \sum_{i' \in \mathsf{C}_l} K_{i'} \quad (20)$$

is the regularized prototype of the winner neuron $\varphi(i)$, that could be seens as a weighted sum of the prototypes $P_l (l = 1,...L)$ in the influence neighborhood of $\varphi(i)$.

4.2. RTC HEURISTIC

In this section, we will give an algorithm suitable to the RTC's formalism. We consider here the batch SOM: the assignment step maximizes the objective function by considering all prototypes P fixed; representation step maximizes the same function considering the clusters set fixed (the assignment function φ fixed). For a fixed temperature T, the maximization occurs in two alternating phases during successive iterations. We summarize this algorithm in the following points

Step 1. Initialization: Initialize the map C using the Relational Analysis approach

Step 2. Assignment: The $\mathsf{R}^T_{RTC}(\varphi, P)$ is expressed as a sum of independent terms (regularized contributions) and we can replace the both optimization problems by a set of simple equivalent problems. Indeed, $\mathsf{R}^T_{RTC}(\varphi, P)$ can be decomposed in terms of individual contributions of each $i \in I$ in each cell of the map C. It is assumed at this stage that all prototypes are fixed and remains constant by maximizing the function $\mathsf{R}^T_{RTC}(\varphi, P)$ compared to φ. It is easy to see that this maximum is reached for an assignment function defined by:

$$\forall i; \varphi(i) = arg \max_{l} cont^T(K_i, P_l) \quad (21)$$

Step 3. Maximization: The maximization step consist in maximizing the objective function over P by setting the assignment φ in it's constant definition. In others words, maximization step consists in updating each regularized prototype $P_l^T(t)$ of neuron C_l at each iteration t according to the following rule:

$$\forall l;\ P_l^T(t) = \sum_{r=1}^{L} \mathbf{K}^T_{(\delta(r,l))}(t) \sum_{i' \in \mathsf{C}_r(t)} K_{i'} \quad (22)$$

The proposed Batch RTC algorithm is presented in Algorithm2.

Algorithm2: Batch RTC algorithm with a fixed T:

Inputs

C^0 = initial map with L_{max} neurons. N_{iter} = the number of iterations. N = the number of observations. α = the similarity threshold. $\mathbf{K}^T$ = the neighborhood matrix

Initialization: Initialize the map C using RA heuristic

- Run the RA heuristic on the K matrix

- Randomly place the resulting clusters on the map C^0

- Compute the initial prototypes:

$$\forall l; P_l^T(0) \leftarrow \sum_{r=1}^{L_{max}} \mathbf{K}^T_{(\delta(r,l))}(0) \sum_{i' \in \mathsf{C}_r(0)} K_{i'}$$

for t=1 to N_{iter} **do**

for $i = 1$ to N **do** {*Assignment*}

assign the observation i to its closest neuron within the sens of contribution:

$$\varphi_{(i)}(t) = arg \max_{\{l=1,.....,L_{max}\}} cont(K_i, P_l(t-1))$$

end for

for $l = 1$ to L_{max} **do** {*Maximization* }

update prototypes according to

$$P_l^T(t) = \sum_{r=1}^{L_{max}} \mathbf{K}^T_{(\delta(r,l))}(t) \sum_{i' \in \mathsf{C}_r(t)} K_{i'}$$

endfor

endfor

Outputs *a map of L_{max} cells.*

5. AUTOMATIC CLUSTER CHARACTERIZATION THROUGH FEATURES SELECTION FOR CATEGORICAL DATA

Feature selection in clustering must provide features that describe the "best" homogenous cluster. Here, we used the prototype set Pl provided by the RTC algorithm. We then used the variable selection

approach to characterize the resulting clusters associated with cells and group of cells. Thus, to select the relevant features, we use the Scree Test Acceleration Factor (algorithm 4).

To attempt the clustering characterization, we integrate the RTC model and variables selection schema (Scree Test) in one procedure which is presented in the algorithm 3.

Algorithm 3: The automatic clustering characterization Algorithm

Input: Dataset X size $n \times d$

FOR $i = 1$ to n

Build a topological map size C using the RTC algorithm (section 4.2, algorithm 2)

END FOR

FOR $j = 1$ to $|C|$ (for each prototype) Find the relevant subset of features using the ScreeTest procedure (for each cell of the cluster), section 5.2, algorithm 4.

END FOR

OUTPUT: The relevant subset of variables characterizing the C clusters of the map.

5.1. THE COMPLEXITY OF THE CLUSTERING CHARACTERIZATION PROCEDURE

Let N be the number of observations; d – the size of variables and C – the size of the map, the clustering characterization procedure is composed from three phases:

1. Clustering. Using the RTC algorithm, the complexity for this step is $O(C \times N \times d)$;

2. Features selection. The computational time of the Scree Acceleration Test procedure for the C cells (clusters) is: $O(d \log d \times C)$.

So, the total complexity time for the proposed clustering characterization technique is $O(C \times N \times d + C \times d \log d)$. This linear complexity depends on the size of variables which is the case for all the variables selection algorithms, and on the size of the map, because the proposed method uses map prototypes to cluster and to select the relevant features.

5.2. AUTOMATIC VARIABLES SELECTION: CATTELL SCREE TEST

We propose to use an established statistical method, *scree test*, to select the most important features [26].

This statistical test was initially developed to provide a visual technique to select eigenvalues for principal components analysis [26].

The basic idea of Scree Test is to generate, for a principal components analysis (PCA), a curve associated with eigenvalues, allowing random behavior to be identified (a simple line plot). Cattell suggests to find the place where the smooth decrease of eigenvalues appears to level off to the right of the plot. To the right of this point, presumably, one finds only "factorial scree". Non graphical solutions to the Cattell's Scree Test are also proposed [21]: an acceleration factor and the optimal coordinates index. The acceleration factor indicates where the elbow of the scree plot appears. It corresponds to the acceleration of the curve, i.e. the second derivative. Frequently this scree is appearing where the slope of the hill change drastically to generate the scree. It is why many researches choose the criterion eigenvalue where the slope change quickly to determine the number of components for a PCA. It is what Cattell named the elbow. So, they look for the place where the positive acceleration of the curve is at his maximum. In the Cattell's scree method, we can interpret the eigenvalues as the degree of relevance of each factor axis. The concept of covariance or correlation matrix is not appear and is not necessary. Therefore, this method is not specific to PCA or a factorial analysis. Hence, in our case, we use this method to choose variables represented by their prototype vector Pl. The number of variables retained is equal to the number of values preceding this 'scree'. We therefore needed to identify the point of maximum deceleration in the curve.

Figure 1 shows an example of a curve generated using a prototype vector. We observed the scree on the 19th feature which means that the irrelevant features have index values lying in the range $[20-40]$. We used an automated process to apply this technique to each prototypes vector $\mathbf{P}l_j = (Pl_j^1, Pl_j^2, ..., Pl_j^d)$.

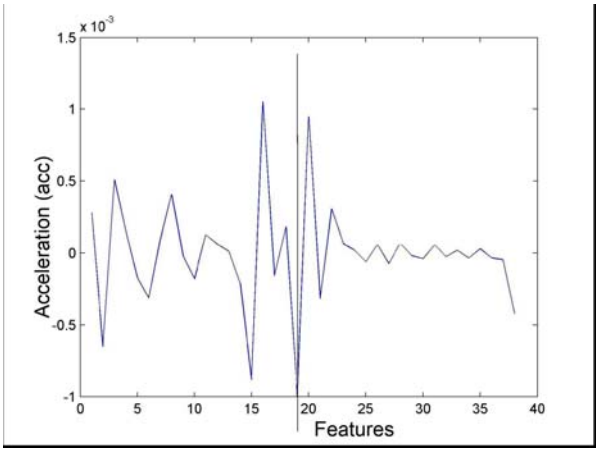

Fig. 1 – An example of the automatic scree test using a prototype vector. The axes X and Y correspond to features and prototype's values, respectively. The scree is indicated by the vertical bar

Thus we have to process the following steps presented in the procedure 1.

Algorithm 4: The Scree Test Acceleration Factor

Input: prototype vector Pl size d

FOR $i = 1$ to d

Sort the vector in descending order $Pl^{[j]}$.

Thus we obtain a new order $Pl^{[j]} = (Pl^{[j],1}, Pl^{[j],2}, ..., Pl^{[j],i}, ..., Pl^{[j],d})$; where i indicates the index order.

END FOR

FOR $j = 1$ to d (on the sorted vector)

Compute the first difference $df_i = Pl^{[j],i} - Pl^{[j],i+1}$ and we obtain the vector $Pl^{[j]}_{df1}$

END FOR

FOR $p = 1$ to d (on the $Pl^{[j]}_{df1}$ vector)

Compute the second difference (acceleration) $acc_i = df_i - df_{i+1}$ obtaining the vector $Pl^{[j]}_{df2}$

END FOR

FOR $l = 1$ to d (on the $Pl^{[j]}_{df2}$ vector)

Find the scree: $\max_i\left(abs(acc_i) + abs(acc_{i+1})\right)$

END FOR

OUTPUT: Retain all the features displayed before the scree (we used the initial index values of features before sorting).

6. EXPERIMENTATIONS AND VALIDATION

6.1. THE DATASETS FOR VALIDATION

In this section, we evaluate the performance of the RTC heuristic on several datasets available at the UC Irvine Machine Learning Repository (Asuncion and Newman, 2007) [1].

Zoo dataset:

This dataset contains 101 animals described with 16 qualitative variables: 15 of the variables are binary and one is numeric with 6 possible values. Each animal is labelled 1 to 7 according to its class.

Nursery Database:

Nursery Database was derived from a hierarchical decision model originally developed to rank applications for nursery schools. The Nursery Database contains examples with the structural information removed, i.e., directly relates nursery to the eight input attributes: parents, hasnurs, form, children, housing, finance, social, health. This dataset has 12960 observations and 8 variables labbeled in 6 classes.

Car Evaluation Database:

Car Evaluation Database was derived from a simple hierarchical decision model originally developed for the demonstration of DEX [28]. The model evaluates cars according their concept structure. The dataset represent a 4 classes problem containing 1728 observations and 6 variables.

Postoperative Patient Data:

The classification task of this dataset is to determine where patients in a postoperative recovery area should be sent to next. Because hypothermia is a significant concern after surgery (Woolery, L. et. al. 1991), the attributes correspond roughly to body temperature measurements. The dataset has 90 observations and 8 variables classified in 3 classes.

SPECTF heart dataset:

The dataset describes diagnosing of cardiac Single Proton Emission Computed Tomography (SPECT) images. Each of the patients is classified into two categories: normal and abnormal. The database of 267 SPECT image sets (patients) was processed to extract features that summarize the original SPECT images. As a result, 44 continuous feature pattern was created for each patient. The pattern was further processed to obtain 22 binary feature patterns. SPECT is a good data set for testing ML algorithms; it has 267 instances that are descibed by 23 binary attributes

6.2. VALIDATION OF THE TOPOLOGICAL ORGANIZATION

There are many ways to measure the accuracy of clustering algorithm. One of the ways of measuring the quality of a clustering solution is the cluster purity. Let there be L clusters of the dataset I and size of cluster C_l be $|\mathsf{C}_l|$. The purity of this cluster is given by $\text{purity}(\mathsf{C}_l) = \frac{1}{|\mathsf{C}_l|}\max_k(|\mathsf{C}_l|_{cluster=k})$ where $|\mathsf{C}_l|_{cluster=k}$ denote the number of items for the cluster k assigned to cluster l. The overall purity of a clustering solution could be expressed as a weighted sum of individual cluster purities:

$$purity = \sum_{l=1}^{L} \frac{|\mathsf{C}_l|}{|I|} purity(\mathsf{C}_l) \qquad (23)$$

In general, if the values of purity are larger, the clustering solution is better.

6.2.1. RESULTS ON ZOO DATASET

We use the zoo dataset to show the good performance of the RTC algorithm. Using disjunctive coding for the qualitative variable with 6

possible values, the dataset consists of a 101×21 binary data matrix. All 101 observations are used to build a map size 5×5 cells. The learning algorithm provides a prototype for each cell. At the end of the learning phase, each observation, corresponding to an animal, is assigned to the cell with the highest contribution by taking into account the neighborhood relation.

The RTC algorithm starts with the initialization of the grid by distributing the observations using the Relational Analysis approach. The figure 2 shows the class of animals distributed after the initialization step of the RTC algorithm. We use the animals names used in original dataset. To visualize the coherence of the map with the labelling of animals, this figure shows the class number corresponding to each cell after the application of the majority rule in each cell. We remind that during this learning step, the neighborhood information is not considered (the neighborhood function $\mathbf{K}$ is not computed). We can constat that on the initialization grid (figure 2) the observations are not well distributed, there are two set of observations labelled with 7 which are separated by 2 empty cells; we can find also four sets of animals labelled as 1 which are dispersed on the map: two sets on the left top corner, one set is situated on the left bottom corner, and the last one, on the right bottom part of the map. This map demonstrates that the classical RA doesn't use a topological information during the clustering process which could allow a better distribution of the observations.

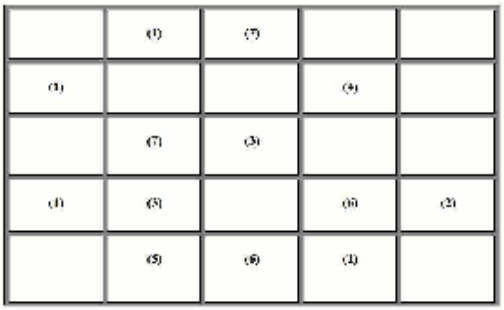

Fig. 2 – Initialization map using Relational Analysis algorithm

After the initialization step, the RTC algorithm will continue the learning process by taking into account the neighborhood relation between all the cells. Figure 3 shows animals names collected by each cell. The map shows that the same class of animals is assigned to cells close to each other.

We can observe that the animals corresponding to the class 1 are clustered in the cells situated on the left bottom of the map (figure 3); the birds which correspond to the class 2 are in the right bottom part of the map. Also, we can analyze that fruitbat from the class 1 is situated nearest to the cell containing the birds (class 2), this is explained that the fruitbat has nearest characterization with the birds even it comes from another family. On the middle of the map there is a cell containing 2 observations from two different classes: the frog (class 5) and penguin (class 2). The RTC algorithm put these two observations in the same cell because the frog and the penguin has very closest specifications even the penguin belongs to birds family and frog, from the amfibia family. Moreover, on the left of this cell there is a cell containing the animals from class 5, and on the right, a cell labelled as class 2. We have the same situation for the cell labelled as 3.5 where the toad and the tortoise has highly correlated features, and the both cells labelled as 5 and 1 are bordered on the right from this cell. The same type of analysis can be applied to the remaining clusters. To give a global view of the homogeneous clustering, we compute the clustering purity for the obtained zoo map and we obtain a purity value of 97.84%.

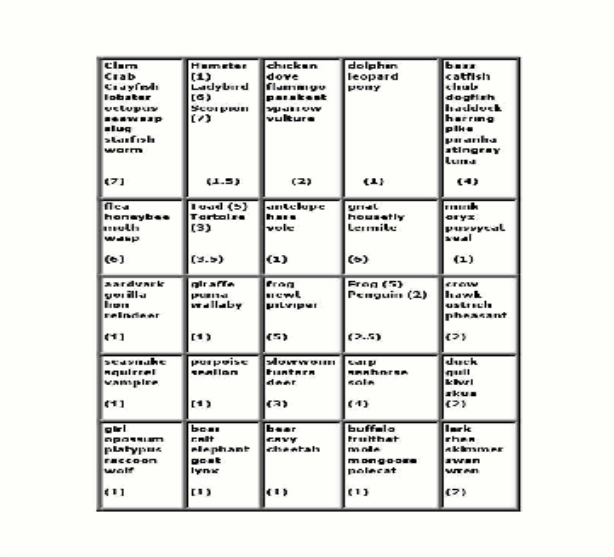

Fig. 3 – Relational Topological Clustering: zoo database

We compare our map with the map obtained using the BeSOM (Bernoulli on Self-Organizing Map) (Lebbah, Rogovschi and Bennani, 2007) which use a probabilistic reformulation of the classical SOM. The map obtained using the BeSOM method is presented in the figure 4. Analyzing both maps obtained with BeSOM (figure 4) and with the proposed RTC approach (figure 3) we can detect some correlations between them: class 5 and 2 are situated in the middle of the map; the majority of the cells containing animals forming the first class are situated on the left bottom corner of the map. Comparing with the BeSOM zoo map, we can observe that RTC zoo map provides more finer cells: in the case of the BeSOM map there are three cells which contains only one observation which respectively will attribute to these ones a 100% of

purity, and 8 cells containing only two observations. The RTC map has no cell which contains only one observation an has only 3 cells with two observations that means that our map has cells with a better distribution of observations.

antelope buffalo deer elephant giraffe hare mole opossum oryx vole (1)	dolphin porpoise (1)	bass catfish chub dogfish herring pike piranha stingray tuna (4)	carp haddock seahorse sole (4)	clam seawasp (7)
aardvark bear boar cheetah leopard lion lynx mink mongoose polecat puma pussycat raccoon wolf (1)	frog frog newt toad tuatara (5)	pitviper slowworm (3)	slug worm (7)	crab crayfish lobster octopus starfish (7)
calf cavy goat hamster pony reindeer (1)	scorpion (7)	kiwi ostrich penguin rhea vulture (2)	girl seal sealion (1)	platypus seasnake tortoise (3)
fruitbat squirrel vampire (1)	duck flamingo swan (2)	chicken dove lark parakeet pheasant sparrow wren (2)	flea termite (6)	gnat (6)
gorilla wallaby (1)	crow gull hawk skimmer skua (2)	honeybee wasp (6)	housefly moth (6)	Ladybird (6)

Fig. 4 – BeSOM map

In order to show the good performance of the Relational Topological Clustering (RTC) approach we use several binary datasets of different sizes. For each dataset we learned a map of different size (from 4x4 to 10x10) and we indicate in the table 1 the purity of clustering after the first iteration using the classical RA and the map purity at the end of the learning process with the RTC technique. The results illustrate that the proposed technique increase the purity index compared to the classical RA and allows to obtain a topological map by computing the neighborhood function between the cells. The obtained topological result (the map) allows the visualization and the analysis of the clustering result.

Table 1. Experimentation results on different datasets using RTC approach

DB	DB size	Map size	RA purity	RTC purity
Zoo	101x17	5x5	69.08%	97.84%
Car	1728x6	10x10	70.31%	80.17%
Nursery	12960x8	6x6	50.47%	78.69%
SPECTF	267x22	4x4	57.14%	81.82%
Pos-Operative	90x8	5x5	71.59%	78.21%

6.3. VALIDATION OF FEATURE SELECTION AND CLUSTERING CHARACTERIZATION

6.3.1. ZOO DATASET

We use the zoo dataset to show the good performance of the proposed clustering characterization schema using the RTC algorithm.

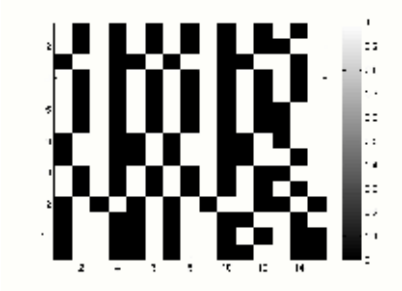

Fig. 5 – A dataset with qualitative variables

The RTC algorithm start with the initialization of the grid by distributing the observations using relational analysis approach. An example of the initial dataset is given in the figure 6, and it is very difficult to detect relevant features when the data contains only binary variables (0 and 1, white and black colors). But, using our proposed Clustering Characterization procedure which allows the dimensionality reduction of the dataset, we are able to construct a prototype matrix which represents the neurons from the RTC map. This matrix contains only continuous features as it is shown in the figure 6 where the red (darkest) color corresponds to the most relevant features for the respective neuron and the blue (white) color – to the noisy features.

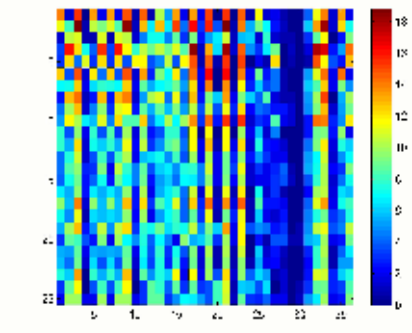

Fig. 6 – A prototype matrix: zoo map

Using Scree Test technique for the RTC map, we will select relevant features for each cell; and we give an example of four clusters from this map: cell 1, 7, 22 and 24. The neuron 1 captured the following samples (animals): bear, boar, cheetah, leopard, lion, lynx, mole, mongoose, polecat, pussycat, raccoon.

The housefly, moth and wasp characterize the 7th cell, and the neuron 22 contains: clam, crab, crayfish, lobster, starfish. Finally, the 24 micro-cluster captured these animals: frog, newt, pitviper and tuatara.

The selected features for these four cells are given in the Table 2, where 0 shows the absence of the corresponding variable (the '0' modality), and 1 – the presence of the variable. These selected features are the most relevant for each neuron which characterize each cell. These results can be easily validated by analyzing the table 2 from a zoological/biological point of view.

Table 2. Selected features on the zoo map

Zoo	selected features	means
cell 1	2(1),11(1),6(0), 5(1),13(0), 9(1)	hair, breathes, airborne, milk, fins, toothed
cell 7	11(1), 12(1), 3(0), 6(1), 10(0)	breathes, venomous, feathers, airborne, backbone
cell 22	13 (0), 14 (5), 3 (0), 6 (0)	fins, legs, feathers, airborne
cell 24	3 (0), 6 (0)	Feathers, airborne

In order to validate the proposed method we compute also the accuracy map index before and after feature selection. For the map (size 5x5) obtained from zoo dataset the accuracy index is 89,15. The noisy features founded by the Scree Test algorithm are the features 29 and 30. We eliminate these variables from the dataset and we re-build the map and the new accuracy index increase to 95,65.

To analyze the impact of the map size on the results we build another map size 4x4, and the new accuracy index before feature selection is 84,16 but this time the noisy features are 4, 22, 29, 30, and the accuracy index after feature selection increase to 88,54.

By analyzing these indexes we can conclude that after feature selection the purity of the map are better and the smaller is the map, the biger is the number of the eliminated features, but the accuracy index are smaller when the map size decrease.

6.4. RESULTS FOR OTHER DATASETS

We tested our proposed algorithm on additional datasets with different characteristics. For the proposed method, we show in Table 3 the feature selection results obtained for the SPECTF, Nursery and Pos-Operative datasets. We will not discuss these results as it is difficult to evaluate the quality of the selected features for each cell when the intersect of the selected features for all the cells are the total number of the initial variables. The aim of these results is to show that for various datasets of different sizes we can cluster, and characterize the cells (clusters) in an automatic way.

Table 3. Selected features on the zoo map

Dataset	Map size	Nb of selected features for each cell
SPECTF	3x3	703, 1, 663, 864,
80x1127		1011, 856, 760, 808, 772
Nursery	4x4	15, 7, 3, 19, 14, 23, 2, 16,
12960x29		7, 12, 15, 14, 12, 8, 14, 11
Pos-Operative	3x3	21, 15, 12, 7, 3,
90x24		8, 19, 20, 17

7. CONCLUSION

We have proposed in this paper a new automatic learning model using the Relational Topological approach for multidimensional categorical data clustering and visualization, inspired from the SOM principle and the Relational Analysis formalism. We have also proposed a process for dimensionality reduction using features selection in the unsupervised learning paradigm in an automatic way. This process uses the RTC algorithm to learn and to build a self-organizing map from a categorical dataset. Several experiments are given and our proposed approach demonstrated the efficiency for simultaneous clustering and feature selection. For future work, we will propose an extended model which will be able to escape the α parameter required by RA and RTC algorithms, and to validate the clustering characterization (feature selection for each cell) using the computed contributions during the learning step.

8. REFERENCES

[1] A. Asuncion and D.J. Newman. *UCI Machine Learning Repository.* [http://www.ics. uci.edu/ mlearn/MLRepository.html]. Irvine, CA: University of California, School of Information and Computer Science. (2007).

[2] Barbara Hammer, Alexander Hasenfuss, Fabrice Rossi and Marc Strickert. Topographic Processing of Relational Data. *In Proceedings of the 6th Workshop on Self-Organizing Maps*

(WSOM 07), Bielefeld, Germany. September 2007.

[3] M. Cottrell and P. Letremy. Analyzing surveys using the Kohonen algorithm. *Proc. ESANN 2003,* Bruges, 2003, M.Verleysen Ed., Editions D Facto, Bruxelles, pp. 85-92.

[4] E.W. Forgy. Cluster analysis of multivariate data: efficiency versus interpretability of classification. *in Biometrics,* (vol. 21), (1965), pp. 768-780.

[5] T. Kohonen. Self-Organizing Maps. *Springer Series in Information Sciences*, vol 30, Springer.

[7] M. Lebbah, F. Badran and S. Thiria. Topological map for binary data. *in ESANN,* 2000.

[8] M. Lebbah, N. Rogovschi and Y. Bennani. BeSOM: Bernoulli on Self-Organizing Map. *in International Joint Conference on Neural networks, IJCNN,* August 2007.

[9] M. Lebbah, Y. Bennani and N. Rogovschi. A Probabilistic Self-Organizing Map for Binary Data Topographic Clustering. *in International Journal of Computational Intteligence and Applications,* World Scientific Publishing Compagny. (Vol. 7), (No. 4), (2008). Pp. 363-383.

[10] F. Leich, A. Weingessel and E. Dimitriadou. Competitive Learning for Binary Data. *in Proc of ICANN'98*, septembre 2-4. Springer Verlag, 1998.

[11] P. Letremy. Traitement de données qualitatives par des algorithmes fondés sur l'algorithme de Kohonen. *SAMOS-MATISSE UMR 8595*, UniversiteS de Paris 1, 2005.

[12] J.F. Marcotorchino. Relational analysis theory as a general approach to data analysis and data fusion, *in Cognitive Systems with interactive sensors*, 2006.

[13] J.F. Marcotorchino, P. Michaud. Optimisation en analyse ordinale des données. (*In Masson*, 1978.)

[14] J.F. Marcotorchino. L'analyse factorielle relationnelle: partie I et II. *Etude du CEMAP, IBM France*, (vol. MAP-03), (décembre 1991).

[15] J.F. Marcotorchino. Dualité Burt-Condorcet: relation entre analyse factorielle des correspondances et analyse relationnelle. (*Etude du CEMAP, IBM France*, in l'analyse des correspondances et les techniques connexes. Springer 2000.)

[16] Zighed D. A, Hacid H., Aupetit M. Topological Learning. *Proceedings of Toplearn workshop of ISMIS*,Prague, 2009.

[17] Grozavu N., Bennani Y. and M. Lebbah. From variable weighting to cluster characterization in topographic unsupervised learning. *IJCNN'09: Proceedings of the 2009 international joint conference on Neural Networks*, ISBN 978-1-4244-3549-4, pages 609-614, Atlanta, Georgia, USA.

[18] Guérif S. and Y. Bennani. Dimensionality reduction trough unsupervised features selection. *International Conference on Engineering Applications of Neural Networks*, Hellas, (2007).

[19] L. Labiod, N. Grozavu and Y. Bennani. Relational Topological Clustering. *WCCI 2010 IEEE World Congress on Computational Intelligence, IJCNN'10*, July, 18-23, 2010 – CCIB, Barcelona, Spain. pp. 3493-3500.

[20] A. John Lee and Michel Verleysen. Unsupervised Dimensionality Reduction: Overview and Recent Advances. *WCCI 2010 IEEE World Congress on Computational Intelligence, IJCNN'10*, July, 18-23, 2010 – CCIB, Barcelona, Spain. pp. 4163-4170.

[21] G. Raiche, M. Riopel and J.G. Blais. Non graphical solutions for the Cattell's scree test. *In International Meeting of the Psychometric Society, IMPS 2006*, HEC, Montreal, 2006.

[22] Asim Roy, 2010, On NSF "open questions," Some External Properties of the Brain as a Learning System and An Architecture for Autonomous Learning. *WCCI 2010 IEEE World Congress on Computational Intelligence, IJCNN'10*, July, 18-23, 2010 – CCIB, Barcelona, Spain, p.3159-3166.

[23] M. Strickert, N. Sreenivasulu, S. Peterek, W. Weschke, H.P. Mock and U. Seiffert. Unsupervised Feature Selection for Biomarker Identification in Chromatography and Gene Expression Data. *In ANNPR*, (2006), pp. 274-285.

[24] John G. Taylor. A Roadmap for Autonomous Adaptive Systems: The Brain-Guided Attention (BGA) System. *WCCI 2010 IEEE World Congress on Computational Intelligence, IJCNN'10*, July, 18-23, 2010 – CCIB, Barcelona, Spain, pp. 412-419.

[25] N. Wiratunga, R. Lothian and S. Massie. Unsupervised Feature Selection for Text Data. *In ECCBR, Lecture Notes in Computer Science*, (v. 4106) (2006) pp. 340-354.

[26] R. Cattell. The scree test for the number of factors. *Multivariate Behavioral Research*, 1: 245276, 1966.

[27] Bennani Y. Adaptive weighting of pattern features during learning. *International Joint Conference on Neural Networks, IEEE – IJCNN'99.* (1999)

[28] M. Bohanec, V. Rajkovic. Expert system for decision making. *Sistemica* 1(1) (1990) pp. 145-157.

Lazhar Labiod *received his B.S. degree in statistics in 2000 at the National Institute of statistics, Algeirs. He received a Master 2 Research degree in Statistics at the Paris 6 university in 2003. In 2008 he obtained the PhD degree in Statistics at the Paris 6 University, France. Now, Lazhar Labiod is Postdoctoral fellow at the Institut Galilée, Paris 13 University, where he works in the 'Machine Learning & Applications (A3)' research team. Lazhar labiod's main research focuses on machine learning, statistics, clustering and unsupervised topographic learning.*

Nistor Grozavu *received his B.S. degree in Informatics in 2005 at the Technical University of Moldova. He received a Master 2 Research degree in 'Fundamental Computer Science' at the Université de la Mediterrannée, Marseille, France in 2006. In 2009 he obtained the PhD degree in the 'Computer Science' at the Paris 13 University, France. Now, Nistor Grozavu is assistant professor at the Institut Galilée, Paris 13 University, where he works in the 'Machine Learning & Applications (A3)' research team. Nistor Grozavu's research interests are: unsupervised topological learning, dimensionality reduction, collaborative learning, fusion, clustering. He is also member of the IEEE, INNS, SFDS, EGC and AML group.*

Younès Bennani *received B.S. degree in Mathematics and Computer Science from Rouen University, in 1987. Subsequently, he received the M.Sc. and the Ph.D. degree in Computer Science from The University of Paris 11, Orsay, in 1988 and 1992, respectively, and the "Habilitation à Diriger des Recherches" (Accreditation to lead research) degree in Computer Science from the Paris 13 University in 1998. Dr. Younès Bennani joined the Computer Science Laboratory of Paris-Nord (LIPN-CNRS) at Paris 13 University in 1993 as Assistant Professor. In 2001, he was appointed to a Full Professor of computer science in the Paris 13 University.*

Prof. Dr. Younès Bennani research interests are in theory of Connectionist Learning (Neural Networks), Statistical Pattern Recognition and Datamining. He is also interested in the application of these models to speech/speaker/languages/-images recognition, diagnosis of complexe systems, users modelling, webmining and call mining.

Prof. Dr. Younès Bennani's areas of expertise are unsupervised learning, cluster analysis, dimensionality reduction, features selection, features construction, data visualisation, and large-scale data mining. He has published 2 books and approximately 150 papers in refereed conferences procedings or journals or as contributions in books. Prof. Dr. Younès Bennani is the head of the Machine Learning research team of the LIPN-CNRS Labs. He gives the MSc lecture on machine learning, data mining and statistical pattern recognition at the Paris 13 University.