computing@computingonline.net
www.computingonline.net

ISSN 1727-6209
International Journal of Computing

A SELF-ORGANIZING MAP FOR MIXED CONTINUOUS AND CATEGORICAL DATA

Nicoleta Rogovschi [1)], Mustapha Lebbah [2)], Younès Bennani [2)]

[1)] LIPADE – University Paris Descartes
45, rue des Saints Pères
75270 Paris Cedex 06, France
nicoleta.rogovschi@parisdescartes.fr
[2)] LIPN – UMR CNRS 7030 Institut Galilée – University Paris-Nord
99, avenue Jean-Baptiste Clément 93430 Villetaneuse, France
{mustapha.lebbah, younes.bennani}@lipn.univ-paris13.fr

Abstract: *Most traditional clustering algorithms are limited to handle data sets that contain either continuous or categorical variables. However data sets with mixed types of variables are commonly used in data mining field. In this paper we introduce a weighted self-organizing map for clustering, analysis and visualization mixed data (continuous/binary). The learning of weights and prototypes is done in a simultaneous manner assuring an optimized data clustering. More variables has a high weight, more the clustering algorithm will take into account the informations transmitted by these variables. The learning of these topological maps is combined with a weighting process of different variables by computing weights which influence the quality of clustering. We illustrate the power of this method with data sets taken from a public data set repository: a handwritten digit data set, Zoo data set and other three mixed data sets. The results show a good quality of the topological ordering and homogenous clustering.*

Keywords: *Self-organizing map (SOM), unsupervised learning, continuous and categorical data.*

1. INTRODUCTION

Large quantity of mixed data, containing continuous, categorical variables, is commonly used in modern data sets. Most of clustering algorithms assume that all variables are either continuous or categorical. Note that categorical variable is ordinal or nominal encoded using the binary coding. When mixed-type data are encountered, some data preprocess is performed to convert the inappropriate types of variables to the desired type prior to the application of the algorithms.

Visualization is an advantageous feature in terms of data mining, especially, in the initial data exploration stage. In this paper, we present an approach for visually analyzing multivariate mixed data. The proposed approach is based on an extended self-organizing map methods. The topological map proposed by Kohonen [12] uses a self-organization algorithm (SOM) which provides quantization and clustering of the observation space. More recently, new models of topological maps dedicated to specific data were proposed in [3, 10, 15, 14]. Some of these models are based on a probabilistic formalism and the others are quantization methods. Like the conventional self-organizing map (SOM), the extended SOM can project high dimensional data to a lower-dimensional space for visual inspection. Most previous clustering algorithms focus on continuous or binary data. As pointed in many papers, continuous data clustering are not appropriate for categorical variable and vice-versa. There are also not suitable algorithms for the task of clustering mixed data. Especially in this work we are interested by clustering model using weighting approach that involves numerical value to each type of variable. It allows us to give information about the relevance of the variable. Thus, variables with strong weight are relevant and has participated actively in the process of clustering. In recent years, more attention has been paid to clustering mixed variable using weighting approaches. Reducing the dimensionality of high-dimensional mixed data is beneficial for visualization and also is an important preprocessing step for many problems in machine learning and statistical pattern recognition. In the literature there are approaches based on weighting as [8, 4, 7, 5]. For the continuous data, a model for local variables weighting using SOM was proposed, called *lw*-SOM [6]. This algorithm is an adaptation to SOM of

the weighting approach proposed for K-means by [8]. The model lw-SOM is dedicated to continuous variables and is not directly applicable to categorical data. In this paper we propose a topological self-organizing algorithm for analyzing mixed variables (continuous and categorical encoded with binary coding). It is a quantization model which provides a set of interpreted prototypes. The variable weights provide to a user the relevance and the degree of each variable for the clustering process.

The remaining of the article is organized as follows. In section 2, we present the model and the iterative algorithm. In the section 3, we present some applications of proposed method. The experiments concern handwritten numerals $(0-9)$, and three other data sets available in [2]. These data sets allow us to prove the importance of the weighting for the clustering process. Our conclusions are reported in section 4.

2. LOCAL WEIGHTED MIXED TOPOLOGICAL MAP

To enable the analysis of mixed data (categorical and continuous data, we present lw-MTM which is based on Self-Organizing Map model.

As with traditional self-organizing map, we assume that the lattice C has a discrete topology (discrete output space) defined by an indirect graph. Usually, this graph is a regular grid in one or two dimensions. We denote the number of cells in C as N_{cell}. For each pair of cells (i, j) on the map, the distance $\delta(i, j)$ is defined as the length of the shortest chain linking cells i and j. The lw-MTM (Local Weighted Mixed Topological Map) model is based on the quantization formalism of topological maps.

Let A be the learning data set, where each observation $\mathbf{x} = (x^1, x^2, ..., x^k, ..., x^d)$ is made of two parts: continuous part $\mathbf{x}^{r[.]} = (x^{r[1]}, x^{r[2]}, ..., x^{r[n]})$ $(\mathbf{x}^{r[.]} \in R^n)$ and categorical part $\mathbf{x}^{c[.]} = (x^{c[1]}, x^{c[2]}, ..., x^{c[l]}, ..., x^{c[k]})$ where the l^{th} component $x^{c[l]}$ have M_l modalities. Each categorical variable can be coded with a binary variable. Thus, each categorical variable $x^{c[l]}$, is coded with the vector $x^{b[.]} = (x^{b[1]}, ..., x^{b[Ml]})$ where $x^{b[l]} \in \{0,1\} = \beta$. The categorical part can be represented by a binary part $\mathbf{x}^{b[.]} = (x^{b[1]}, x^{b[2]}, ..., x^{b[l]}, ..., x^{b[m]})$ such as each observation $\mathbf{x}$ is thus, a realization of a random variable which belongs to $R^n \times \{0,1\}^m$. Using these notations a particular observation $\mathbf{x} = (\mathbf{x}^{r[.]}, \mathbf{x}^{b[.]})$ is a mixed vector (continuous and binary variables) of dimension $d = n + m$. In our model, we assume that a given data set has been drawn from N_{cell} clusters.

For each cell c of the grid, we associate a prototype vector $\mathbf{w}_\mathbf{c} = (\mathbf{w}_c^{r[.]}, \mathbf{w}_c^{b[.]})$ of dimension d, where $\mathbf{w}_c^r \in R^n$ and $\mathbf{w}_c^{b[.]} \in \beta^m$ which is a binary coding of multidimensional categorical variable $\mathbf{w}_c^{c[.]}$. We denote by W the set of the referents vectors, by W^r the set of the numerical part and by W^b the binary part of the referent vectors.

In the following section we present a new model of topological map dedicated to mixed data. The associated learning algorithm is derived from the batch version of the Kohonen algorithm dedicated to continuous data [13] and the BinBatch algorithm which is dedicated to binary data [15]. This model is improved to take into account the variable weights. In this algorithm, the similarity measure and the estimation of the referent vectors are specific for each type of data: it is the Euclidian distance with the mean vector in the continuous case and the Hamming distance with the median center in the binary case.

2.1. MINIMIZATION OF THE COST FUNCTION

We propose to minimize the following new cost function.

$$G(\phi, W, Y) = \sum_{\mathbf{x} \in A} \sum_{j \in C} K^T(\delta(\phi(\mathbf{x}), j)) \mathbf{y}_j{}^\tau \| \mathbf{x} - \mathbf{w}_j \|^2 \quad (1)$$

where τ is a fitting parameter necessary for estimation of the set of the weight vectors Y, and the function ϕ assigns each observation $\mathbf{x}$ to a single cell in C.

K^T is a neighborhood function depending on the parameter T (called temperature): $\mathsf{K}^T(\delta) = \mathsf{K}(\delta/T)$, where K is a particular kernel function which is positive and symmetric $(\lim_{|x| \to \infty} \mathsf{K}(x) = 0)$. Thus K defines for each cell j a neighborhood region in C. The parameter T allows to control the size of the neighborhood influencing a given cell on the map. As with the Kohonen algorithm, we decrease the value of T between two values T_{max} and T_{min}.

The vector $\mathbf{y}_j = (\mathbf{y}_j^{r[.]}, \mathbf{y}_j^{c[.]})$ is the weighted vector, where $\mathbf{y}_j^{r[.]}$ is the continuous weight part and

$\mathbf{y}_j^{c[.]}$ is a categorical weight variable (not binary variable).

In this expression $\| \mathbf{x} - \mathbf{w}_j \|^2$ is the square of the Euclidian distance. Since for binary vectors the Euclidian distance is no more than the Hamming distance H, then the Euclidian distance can be rewritten by:

$$\| \mathbf{x} - \mathbf{w}_c \|^2 = \| \mathbf{x}^{r[.]} - \mathbf{w}_c^{r[.]} \|^2 + H(\mathbf{x}^{b[.]}, \mathbf{w}_c^{b[.]})$$

Thus, this expression allows to rewrite the cost function as follows:

$$G(\phi, W, Y) = \sum_{\mathbf{x} \in A} \sum_{j \in C} K^T(\delta(\phi(\mathbf{x}), j)) \mathbf{y}_j^{r[.]} D_{euc}(\mathbf{x}^{r[.]}, \mathbf{w}_j^{r[.]}) + \sum_{\mathbf{x} \in A} \sum_{j \in C} K(\delta(\phi(\mathbf{x}), j)) \mathbf{y}_j^{c[.]} H(\mathbf{z}_i^{b[.]}, \mathbf{w}_j^{b[.]})] \quad (2)$$

Where

$$G_{som}(\phi, W, Y) = \sum_{\mathbf{x} \in A} \sum_{j \in C} K^T(\delta(\phi(\mathbf{x}), j)) \mathbf{y}_j^{r[.]} \| \mathbf{x}^{r[.]} - \mathbf{w}_j^{r[.]} \|^2 \quad (3)$$

is the classical cost function used by the weighted Kohonen Batch algorithm [6], and

$$G_{bin}(\phi, W, Y) = \sum_{\mathbf{x} \in A} \sum_{j \in C} K^T(\delta(\phi(\mathbf{x}), j)) \mathbf{y}_j^{b[.]} H(\mathbf{x}^{b[.]}, \mathbf{w}_j^{b[.]}) \quad (4)$$

is the new cost function dedicated to handle categorical variables using binary coding. Hence, in this paper we propose a new cost function to deal with mixed data and we define a new function for binary data.

The minimization of the cost function (1), is made using an iterative process with three steps:

• **1) Assignment step**: assuming that W and Y are fixed, we have to minimize $G(\phi, W, Y)$ with respect to ϕ. This leads to use the following assignment function:

$$\phi(\mathbf{x}) = \arg\min_j \left((\mathbf{y}_j^{r[.]})^\tau \| \mathbf{x}^{r[.]} - \mathbf{w}_j^{r[.]} \|^2 + (\mathbf{y}_j^{c[.]})^\tau H(\mathbf{x}^{b[.]}, \mathbf{w}_j^{b[.]}) \right)$$

• **2) Quantization step**: assuming that ϕ and Y are fixed, this step minimizes $G(\phi, W, Y)$ with respect to W in the space $R^n \times \beta^m$. The minimization of the cost function (1) leads to minimize the function $G_{som}(\phi, W, Y)$ (3) in R^n and $G_{bin}(\phi, W, Y)$ (4) in β^m. It is easy to see that these two minimizations allow to define:

- **the numerical part** $\mathbf{w}_j^{r[.]}$ of the referent vector $\mathbf{w}_j$ as the mean vector as:

$$\mathbf{w}_j^{r[.]} = \frac{\sum_{i \in C} K^T(\delta(i, j)) \sum_{\mathbf{x} \in A, \phi(\mathbf{x}) = i} \mathbf{x}^{r[.]}}{\sum_{i \in C} K^T(\delta(i, j)) n_i}, \quad (5)$$

where n_i represents the corresponding number of assigned observations.

- **the binary part** $\mathbf{w}_j^{b[.]}$ of the referent vector $\mathbf{w}_j$ as the median center of the binary part of the datum $\mathbf{x} \in A$ weighted by $K^T(\delta(j, \phi(\mathbf{x})))$. Each component $\mathbf{w}_j^{b[.]} = (w_j^{b[1]}, ..., w_j^{b[l]}, ..., w_j^{b[m]})$ is then computed as follows:

$$w_j^{b[l]} = \begin{cases} 0 & if \left[\sum_{\mathbf{x} \in A} K^T(\delta(j, \phi(\mathbf{x})))(1 - \mathbf{x}^{b[l]}) \right] \geq \left[\sum_{\mathbf{x} \in A} K^T(\delta(j, \phi(\mathbf{x}))) \mathbf{x}^{b[l]} \right] \\ 1 & otherwise \end{cases} \quad (6)$$

Note that the update for the median center $\mathbf{w}_j$ in our lw-MTM model coincides with the BinBatch model in which each datum $\mathbf{x}$ is weighted proportionally to the neighborhood function centered at the winning prototype for that data vector and evaluated at the prototype $\mathbf{w}_j$.

• **3) Weighting step**: assuming that ϕ and W are fixed, this step minimizes $G(\phi, W, Y)$ with respect to Y in the space R^{n+m}. The weights are computed in the following way:

$$y_j^l = \begin{cases} 0, \, if \; D_j^l = 0 \\ \dfrac{1}{\sum_t \left[\dfrac{D_j^l}{D_t^l} \right]^{\frac{1}{\tau - 1}}}, \, otherwise \end{cases} \quad (7)$$

where

$$D_j^l = \sum_{\mathbf{x} \in A} \sum_{i=1}^{C} K^T(\delta(i, j))(x_i^l - w_j^l)^2$$

The minimization of $G(\phi, W, Y)$ is run by

iteratively performing the three steps. At the end the vector $\mathbf{w}_j$, which shares the same code with the observations can be decoded in the same way, allowing a symbolic interpretation of binary and continuous part of referent vectors.

The nature of the topological model reached at the end of the algorithm, the quality of the clustering and those of the topological order induced by the graph greatly depend on the neighborhood function K. In practice, as for traditional topological map we use a smooth function to control the size of the neighborhood as $K^T(\delta(c,r)) = \exp\left(\frac{-\delta(c,r)}{T}\right)$. Using this kernel function, T becomes a parameter of the model. As in the Kohonen algorithm [13], we repeat the preceding iterations by decreasing T from an initial value T_{max} to a final value T_{min}.

We can define two steps in the operating of the algorithm:

• **The first step** corresponds to high T values. In this case, the influencing neighborhood of each cell i on the map is important and corresponds to higher values of $K^T(\delta(c,r))$. Formulas (5), (6) and (7) use a high number of observations to estimate model parameters. This step provides the topological order.

• **The second step** corresponds to small T values. The number of observations in formulas (5), (6) and (7) is limited. Therefore, the adaptation is very local. The parameters are accurately computed from the local density of the data.

3. EXPERIMENTAL VALIDATIONS

To evaluate the quality of clustering, we adopt the approach of comparing the results to a "ground truth". We use the clustering accuracy to measure the clustering results. This is a common approach in the general area of data clustering. In general, the result of clustering is usually assessed on the basis of some external knowledge about how clusters should be structured. This may imply evaluating separation, density, connectedness, and so on. The only way to assess the usefulness of a clustering result is indirect validation, whereby clusters are applied to the solution of a problem and the correctness is evaluated against objective external knowledge. This procedure is defined by [9] as "validating clustering by extrinsic classification", and has been followed in many other studies [1, 11]. We feel that this approach is the reasonable one if we don't want to judge clustering results by some cluster validity index, which is nothing but a bias toward some preferred cluster property (e.g., compact, or well separated, or connected).

Thus, to adopt this approach we need labeled data sets, where the external (extrinsic) knowledge is the class information provided by labels. Hence, if lw-MTM finds significant clusters in the data, these will be reflected by the distribution of classes. Therefore we operate a vote step for clusters and compare them to the behavior methods from the literature. The so-called vote step consists in the following. For each cluster $c \in \mathsf{C}$:

• Count the number of observations of each class l (call it N_{cl}).

• Count the total number of observation assigned to the cell c (call it N_c).

• Compute the proportion of observations of each class (call it $S_{cl} = N_{cl}/N_c$).

• Assign to the cluster c the label of the most represented class ($l = \arg\max_l(S_{cl})$).

A cluster c for which $S_{cl} = 1$ for some class labeled l is usually termed a "pure" cluster, and a purity measure can be expressed as the percentage of elements of the assigned class in a cluster. The experimental results are then expressed as the fraction of observations falling in clusters which are labeled with a different class from that of the observation. This quantity is expressed as a percentage and termed "error percentage" (indicated as $Err\%$ in the results).

3.1. CATEGORICAL DATA SETS

3.1.1. ZOO DATA

This example is taken from UCI. We use this simple data set to show the good performance of the lw-MTM algorithm. The data set 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. Using disjunctive coding (see Appendix A) for the categorical variable with 6 possible values, defined in table 1, the data set consists of a 101×21 binary data matrix. All 101 animals are used for learning with a map with the dimensions 5×5 cells.

Table 1: Categorical variable with 6 possible values. Each modality is coded in the same way using disjunctive complete coding

Modalities	Binary code
1	1 0 0 0 0 0
2	0 1 0 0 0 0
3	0 0 1 0 0 0
4	0 0 0 1 0 0
5	0 0 0 0 1 0
6	0 0 0 0 0 1

The results of our approach on Zoo data set are presented on figure 1. We can visualize the prototypes and the variables which characterize these prototypes for every cell of the map. Figure 1 shows the animal names collected by each cell. We use the same names used in original data set.

In order to visualize the coherence of the map with animals assigned to each cell, we chose only the variables associated to the modality "yes", which have a weight greater than 0.02. We observe that we have homogeneous groupings which are better separated. We notice that some kinds of fishes are grouped around neighboring cells (cells 12,13,17,19 on the map of the figure 1) with some commons variables: "aquatic", "toothed", "backbone", "tail". The same analysis can be done on the rest of the cells.

3.1.2. HANDWRITTEN DATA

This experiment concerns a data set consisting of the handwritten numerals ("0" – "9") extracted from a collection of Dutch utility maps, UCI. There are 200 samples of each digit such that there is a total of 2000 samples. Each sample is a 15×16 binary pixel image. The data set consisted of a 2000×240 binary data matrix. Each qualitative variable is a pixel with two possible values "On=1" and "Off=0".

The figure 2 shows four maps obtained from the learning of lw-MTM map of 16×16 size with the fitting parameters $\tau = 2$ and $\tau = 3$. In the first column, we can visualize a binary part of the prototypes W^b displayed as an image, where each pixel "black/white" denotes the state of the binary variable ("On/Off"). In the second column, the grey shading shows the relevance of the variables. We observe that these pixels correspond to the contour associated to each image (number). We notice the clear topological organization of the reference images.

We also note that the parameter τ does not need to be great, but it must be as input parameter. In [14] the same data set are used with an algorithm based on mixture models using Bernoulli distribution (BeSOM), which has as parameters binary prototypes and the probability of being different from this prototype for each variable.

Figure 3 shows two maps corresponding to the parameters of mixture models (referents, probabilities). We observe clearly that both approaches (probabilistic and lw-MTM) are able to produce topological maps representing well the data and variables, though lw-MTM determines the parameters in a deterministic manner, with a computational complexity less than BeSOM mixture model.

3.2. MIXED DATA SETS

We use the following three mixed data sets obtained from UCI repository [2].

Heart disease

This is D. Detrano's heart disease data set that was generated by the Clevelande Clinic [2]. The data set has 303 observations, each one is described by 6 continuous and 8 categorical variables. The observations are also classified into two classes, each class is either healthy (buff) or with heart-disease (sick). In both cases we use a binary coding to code a categorical variable. Hence, using a disjunctive coding we obtain $m = 17$ binary variables for Heart disease data set. The variable with two modalities is coded using only one binary variable indicating a presence or absence of modalities. The learning of a map with the dimensions 13×7 cells is made with all observations.

Credit Approval

This file concerns credit card applications. All attribute names and values have been changed to meaningless symbols to protect confidentiality of the data. This dataset is interesting because there is a good mix of attributes – continuous, nominal with small numbers of values, and nominal with larger numbers of values. There are also a few missing values. The data set has 666 observations, each one is described by 9 continuous and 6 categorical variables. Examples represent positive and negative instances of people who were and were not granted credit.

Thyroid disease

This dataset contains thyroid disease records supplied by the Garavan Institute and J. Ross; Quinlan, New South Wales Institute, Syndney, Australia in 1987. The data set has 3163 observations, each one is described by 7 continuous and 12 categorical variables. Five laboratories tests are used to try to predict whether a patient's thyroid to the class hypothyroidism or hyperthyroidism. The diagnosis (the class label) was based on a complete medical record, including anamnesis, scan etc. Table 2, provides a short description of used data sets.

Table 2: Data sets used in the experimentation. #obs: data set size; #cl: number of classes; dim.Cat: categorical dimension; dim.Re: continuous variable dimension

Data sets	dim.Cat	dim.Re	#obs	#cl
Heart disease	8	6	303	2
Credit	6	9	666	2
Thyroid	12	7	3163	2

We use the clustering accuracy for measuring the clustering results. This index is a purity measure which can be expressed as the percentage of elements of the assigned class in a cluster. This is a common approach in the general area of data clustering. We compared the proposed lw-MTM model with similar algorithm MTM without using weight and the probabilistic algorithm PrMTM. We computed the purity index on 50 experiences. The table 3 shows the performances obtained by the proposed model lw-MTM and the other models MTM and PrMTM. We observe an improvement of map purity on all datasets.

Table 3: Comparison of lw-MTM, MTM and PrMTM using the purity index on 50 experimentations. MTM: topological map dedicated to mixed data without using weights. PrMTM: Probabilistic mixed topological map using Gaussian and Bernoulli distributions

% Purity	MTM	PrMTM	lw-MTM
Heart disease (13×7)	83.39	84.45	85.76
Credit (13×10)	82.66	84.57	86.44
Thyroid (21×14)	95.38	97.41	97.53

Analyzing the table 3 we observe for the Heart disease data set, an improvement of the purity index from 83.39% to 85.76% using the same map size. For *Credit* data set, we observe also an improvement of the purity index from 82.66% to 86.44%. Finally with Thyroid data set, we improve the performance from 95.38% to 97.53%. Through to the weights introduced during the learning process, we observe a clear improvement in the purity rate with lw-MTM model.

4. CONCLUSION

In this paper, we proposed a weighted self-organizing map for clustering categorical and mixed data. The weighting of the distance during the learning phase allows to detect the degrees of participation of each variable during the clustering process. More variable has a high weight, more the clustering algorithm will take into account the informations associated to this variable. The weighting distance has the purpose to adapt the (dis)similarity measure between the observations and to improve the clustering results by mainly strengthening the most relevant variables. The weighting distance is very useful in the case of mixed data, because if for the learning data set the categorical part is much larger than the continuous part of the learning data set (and vice versa), the weighting process allows us to regularize the adaptations during the learning phase and to take into account the relevance of each variable. As perspective we can use the computed weights to select the most relevant variables in order to reduce the dimensionality of the data.

5. APPENDIX: GENERAL INFORMATION ABOUT A BINARY DATA

Very often, a binary vector represents a coding of discrete features which have a finite, usually small, number of possible values. Let $\beta^n = \{0,1\}^n$ be a binary data space and $\mathsf{A} = \{\mathbf{x}_i; i = 1,\ldots,N\}$ a set of observations, where each observation $\mathbf{x}_i^{b[.]} = (x_i^{b[1]}, x_i^{b[2]},..., x_i^{b[n]})$ is a binary vector in β^n. Some of these variables, called ordinal variables, have an implicit order, the others are just nominal variables. The general coding used in order to obtain binary data are: (a) *The additive binary coding*: this coding respect the order existing between modalities, (see table 4). (b) *The disjunctive complete coding*: which transforms each nominal feature using the disjunctive coding (see table 4).

Table 4: Coding of modalities

Modalities	Additive coding	Disjunctive coding
1	1 0 0	1 0 0
2	1 1 0	0 1 0
3	1 1 1	0 0 1

Euclidian distance is not adapted to binary data, it is often much more interesting to use an appropriate similarity index. In this paper, we use the Hamming distance H which allows comparison of the binary vectors $\mathbf{x}_m$ and $\mathbf{x}_l$. The Hamming distance measures the number of mismatches between $\mathbf{x}_m^{b[.]}$ and $\mathbf{x}_l^{b[.]}$:

$$H(\mathbf{x}_m^{b[.]}, \mathbf{x}_l^{b[.]}) = \sum_{j=1}^{n} \mathbf{y}^{[j]} \mid x_m^{b[j]} - x_l^{b[j]} \mid \quad (8)$$

The Hamming distance allows the binary median center to be calculated; in this case the most important characteristic in this case of the median

center is a binary vector that has the same interpretation (same coding) as the observations in the data space.

For the following, we assume that each $\mathbf{x}_i^{b[.]}$ is taken with corresponding weight γ_i. By definition the median center of A is any point $\mathbf{w} = (w^{b[1]}, w^{b[2]}, ..., w^{b[n]})$ included in β^n minimizing the inertia of A:

$$I(\mathbf{w}) = \sum_{i=1}^{I} \gamma_i \mathbf{y}^{b[.]} H(\mathbf{x}_i^{b[.]}, \mathbf{w}^{b[.]})$$

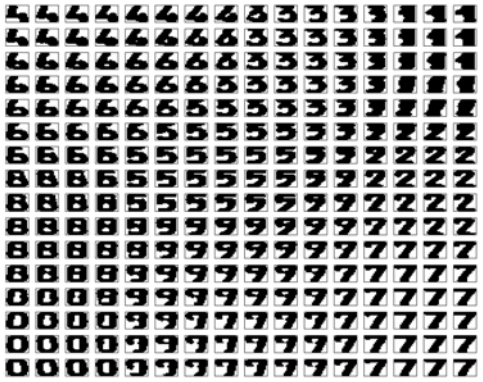
Each component $w^{b[j]}$ minimizes $I(w^{b[j]}) = \sum_{i=1}^{N} \gamma_i y^b[j] \mid x_i^{b[j]} - w^{b[j]} \mid$ which can be rewritten as:

$$I(w^{b[j]}) = y^{b[j]} \left(w^{b[j]} \Gamma_0 + (1 - w^{b[j]}) \Gamma_1 \right),$$

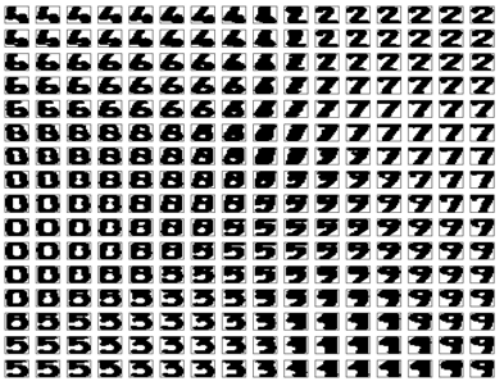
where $\Gamma_0 = \sum_{i=1}^{N} \gamma_i (1 - x_i^{b[j]})$, represents the sum of weighted observations which the value is equal to 0, and $\Gamma_1 = \sum_{i=1}^{N} \gamma_i x_i^{b[j]}$ represents the sum of weighted observations which the value is equal to 1. Thus, to find the median value w^j which minimizes $I(w^{b[j]})$, we select $w^{b[j]} \in \{0,1\}$ so that $I(w^{b[j]})$ have minimum value Γ_0 or Γ_1. Hence, we take $w^{b[j]} = 1$ if $\Gamma_1 > \Gamma_0$ and $w^{b[j]} = 0$ if $\Gamma_1 < \Gamma_0$. This rule is simplified when the weights are identical for all variables: $w^{b[j]}$ is the value 0 or 1 most often chosen by the observations on the binary variable j.

case 1 : boar,calf,cheetah,goat, leopard,lion,lynx,mongoose, polecat,pony,puma, pussycat,raccoon, reindeer, tortoise, wolf *hair, milk,predator,toothed,tail,catsize*	*case 2 :* aardvark,bear, cavy,hamster *hair,milk,toothed,backbone,breathes, tail,*	*case 3* *eggs,toothed, backbone,breathes,tail*	*case 4 :* lark,pheasant,sparrow, *feathers,eggs,airborne backbone, breathes,tail*	*case 5:* *feathers,eggs,tail,backbone*
case 6: girl *hair,milk,predator,toothed,backbone tail,catsize*	*case 7:* *eggs,aquatic,predator,toothed, legs,tail,catsize,backbone,breathes, fins*	*Case8:* haddock,newt,penguin, seahorse, sole *feathers,eggs,aquatic, backbone,breathes,tail*	*Case9:* carp,chicken,crow,dove parakeet,rhea,skimmer, duck, flamingo,gull, hawk, skua,swan,vulture kiwi,ostrich, *eathers,airborne backbone,breathes tail*	*case 10:* *feathers,eggs ,airborne, predator,backbonebreathes, tail*
case 11: *milk,aquatic,predator,toothed backbone,breathes,tail, catsize*	*case 12:* dogfish,dolphin,pike platypus,porpoise,tuna *eggs,aquatic,predator toothed,backbone,fins legs,tail,catsize*	*case 13:* bass,catfish,chub,herring, piranha, scorpion,seasnake,stingray *eggs,aquatic, predator, toothed backbone, fins,legs,tail*	*case 14:* clam, gnat,octopus pitviper,seawasp slowworm,tuatara *eggs,predator,backbone breathes, tail*	*case 15:* lobster,starfish crab,crayfish *eggs,aquatic,predator*
case 16: hare, squirrel, vole *hair,milk,toothed,backbone tail,*	*case 17:* frog, fruitbat, vampire *hair,milk,predator,toothed, backbone,breathes,tail*	*case 18:* *eggs,aquatic, predator, toothed,backbone, breathes, tail*	*case 19:* honeybee, housefly, moth, slug, wasp, worm *eggs,breathes*	*case 20* *eggs, breathes*
case 21: antelope, buffalo, deer,elephant giraffe, gorilla, oryx, seal, wallaby *tail, catsize*	*Case 22:* mink,mole, opossum,sealion *hair, milk,tail,catsize*	*case 23:* frog,toad *eggs,aquatic,predator toothed,backbone, breathes*	*case 24:* flea, termite *eggs.*	*case 25:* ladybird *eggs*

Fig. 1 – 5×5 lw -MTM map. The examples are presented in each cell followed by the relevant variables shown in red

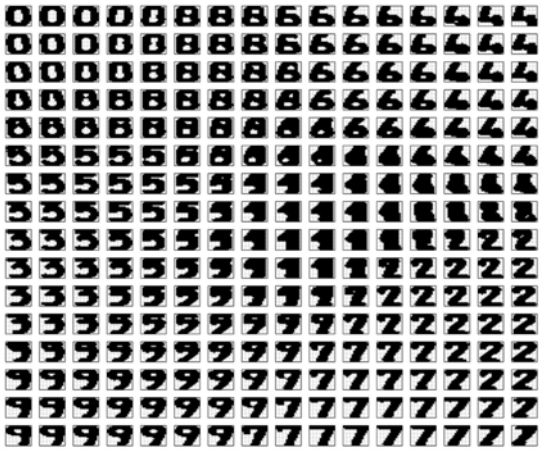


$\tau = 2$



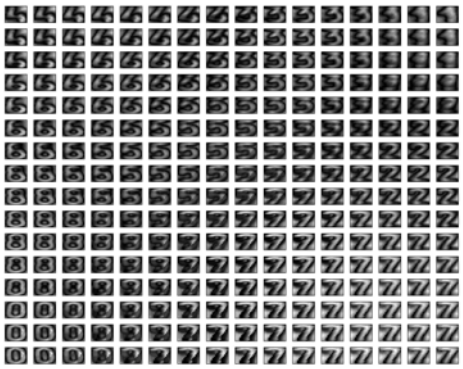
$\tau = 3$

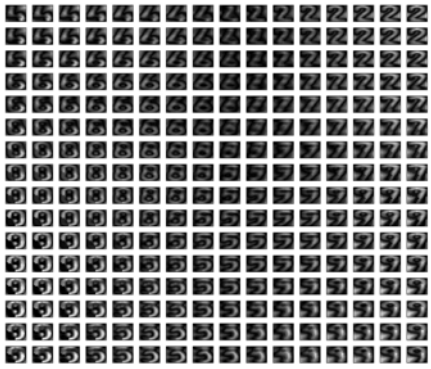
Fig. 2 – The map lw-MTM 16×16 representing the set of prototypes W and weights Y ($\tau = 2, \tau = 3$)

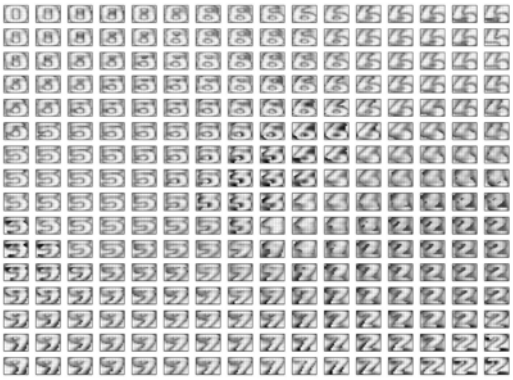


(*W*)

Fig. 3 – The BeSOM map 16×16. The map on the left represents the binary prototype and the map on the write the probability to be different from the prototype







(*Proba*)

6. REFERENCES

[1] B. Andreopoulos, A. An, and X. Wang. Bi-level clustering of mixed categorical and numerical biomedical data. *International Journal of Data Mining and Bioinformatics*, 1(1) (2006) pp. 19-56.

[2] A. Asuncion and D. Newman. *UCI machine learning repository*. http://www.ics.uci.edu/-mlearn/MLRepository.html, 2007.

[3] C.M. Bishop, M. Svensén, and C.K.I. Williams. GTM: The generative topographic mapping. *Neural Comput*, 10(1) (1998) pp. 215-234.

[4] A. Blansche, P. Gancarski, and J. Korczak. Maclaw: A modular approach for clustering with local attribute weighting. *Pattern Recognition Letters*, 27(11) (2006) pp. 1299-

1306.
[5] N. Grozavu, Y. Bennani, and M. Lebbah. Pondération locale des variables en apprentissage numérique non-supervisé. Sophia-Antipolis, France, (2008) pp. 45-54.
[6] N. Grozavu, Y. Bennani, and M. Lebbah. From variable weighting to cluster characterization in topographic unsupervised learning. *In IJCNN'09: Proceedings of the 2009 international joint conference on Neural Networks*, Institute of Electrical and Electronics Engineers Inc., The, 2009 pp. 609-614.
[7] S. Guérif and Y. Bennani. Dimensionality reduction trough unsupervised features selection. *International Conference on Engineering Applications of Neural Networks*, 2007.
[8] J. Z. Huang, M. K. Ng, H. Rong, and Z. Li. Automated variable weighting in k-means type clustering. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 27(5) (2005) pp. 657-668.
[9] A. K. Jain and R. C. Dubes. *Algorithms for Clustering Data*. Prentice-Hall, Inc., Upper Saddle River, NJ, USA, 1988.
[10] A. Kaban and M. Girolami. A combined latent class and trait model for the analysis and visualization of discrete data. *IEEE Trans. Pattern Anal. Mach. Intell*, (23) (2001) pp. 859-872.
[11] S. S. Khan and S. Kant. Computation of initial modes for k-modes clustering algorithm using evidence accumulation. *In IJCAI*, (2007) pp. 2784-2789.
[12] T. Kohonen. Self-organizing Maps. Springer Berlin, Vol. 30, Springer, Berlin, Heidelberg, New York, 1995, 1997, 2001. Third Extended Edition, 2001. 501 p.
[13] T. Kohonen. Self-organizing Maps. Springer Berlin, 2001.
[14] M. Lebbah, Y. Bennani, and N. Rogovschi. A probabilistic self-organizing map for binary data topographic clustering. *International Journal of Computational Intelligence and Applications*, 7(4) (2008) pp. 363-383.
[15] M. Lebbah, S. Thiria, and F. Badran. Topological map for binary data. *In Proceedings European Symposium on Artificial Neural Networks-ESANN*, Bruges, April 26-27-28, 2000, pp. 267-272.

Nicoleta Rogovschi *received her B.S. degree in Informatics in 2005 at the Technical University of Moldova. She received a Master 2 Research degree in 'Computer Science' at the Paris 13 University, France in 2006. In 2009 she obtained the PhD degree in 'Computer Science' at the Paris 13 University, France. Now she is associate professor at the Paris Descartes University, where she works in the 'Data Mining& Machine Learning(GFD)' research team. Her research interests are: unsupervised learning, mixture models, Markov chains, clustering.*

Mustapha Lebbah *is currently Associate Professor at the University of Paris 13 and a member of Machine learning Team A3, LIPN. His main researches are centred on machine learning (Self-organizing map, Probabilistic and Statistic, unsupervised learning, cluster analysis. Graduated from USTO University where he received his engineer diploma in 1998. Thereafter, he gained an MSC (DEA) in Artificial Intelligence from the Paris 13 University in 1999. In 2003, after three year in RENAULT R&D, he received his PhD degree in Computer Science from the University of Versaille. 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.