# Feature Selection for Self-Organizing Map

Khalid Benabdeslem[1] and Mustapha Lebbah[2]

[1] University of Lyon1, LIESP,
8, Boulevard Niels Bohr, 69622 Villeurbanne, Cedex, France
kbenabde@bat710.univ-lyon1.fr

[2] University of Paris13, LIM&BIO, UFR de Santé,
Médecine et Biologie Humaine (SMBH) - Léonard de Vinci
74, rue Marcel Cachin 93017, Bobigny Cedex, France
lebbah@limbio-paris13.org

**Abstract**. *In this paper, we present a new heuristic measure for optimizing database used as input layer of Self Organizing Map (SOM). This heuristic called HI-SOM (Heuristic Input for SOM) consists of selection of variables for clustering with SOM algorithm. HI-SOM allows to identify and to select important variables in the feature spaces. Thus, we eliminate redundant variables and those do not contain enough relevant information. The proposed measure is used in SOM learning algorithm in order to reduce the database dimension. Hence, HI-SOM select the important variables to train the "best" SOM. We illustrate this method with three databases from public data set repository. We show the effectiveness to identify the important variables which gives homogenous clusters.*

**Keywords.** SOM, Selection of variables, Clustering

## 1. Introduction

Clustering algorithms seek to automatically discover underlying patterns in a dataset. To this end, they conduct a search through the space of possible organizations of data, preferring those group similar instances together and keep dissimilar instances separately [15, 16]. These clustering algorithms are generally used in unsupervised fashion. They are presented with a set of data instances that should be grouped according to some notion of similarity.

Nowadays, to solve learning from examples problem, a large amount of data can be available due to the proliferation of acquisition systems. Hence, more and more features characterize each example. Feature selection algorithm addresses the problem of selection of input variables that give best model representing the problem [12, 13, 14].

The collected data required for clustering, are not necessarily informative at all. Therefore, this data can be noised, non significant or non pertinent for the required task. The features selection is thus, an important task which allows reducing dimensions of database and facilitating the learning process.

In this paper, we propose a heuristic measure that optimizes the number of features (variables) presented as input data to SOM algorithm. This measure aims to improve the clustering. Therefore, this improvement is insured by selecting in iterative process, the model which improves the quality of homogenous clustering and data set topology representation accuracy.

The remainder of the paper is organized as follows: firstly, we will briefly present the general concept of SOM; secondly, we explain the proposed heuristic measure. Finally, we give the experiment results using two real databases.

## 2. SOM algorithm

The self organizing map (SOM) introduced by Kohonen [8] has been used for visualization and clustering high-dimensional databases. Visualization as proposed on SOM methods uses deformable discrete lattice to translate data similarities into special relationships. A large variety of related algorithms have been derived from the first SOM model, which differs from one to another but share the same idea that introduces the topological order between the different clusters. The standard self-organizing

map (SOM), consist of discrete topology defined as an undirected graph noted *C*, ( Figure. 1).

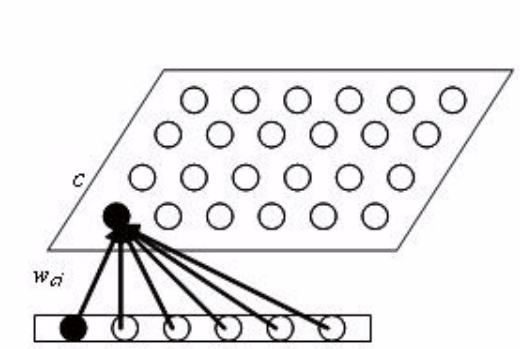



**Figure 1. Importance of variable *j* over the neuron: *c***

Usually this graph is a regular grid in one or two dimensions with $N_c$ neurons. For each pair of neurons $(c,r)$ on the grid, the distance $\delta(r,c)$ is defined as being the shortest path on the graph between *c* and *r*. This discrete distance defines a neighbourhood with positive kernel function $K^T(\delta)$ parameterized by the parameter *T* which controls the neighbourhood size of each neuron *c*.

Let $A=\{z_i, .i=1..N\}$ the learning data set where $z_i \in R^d$ $(z_i=(z_{i1}..., z_{ij}, ....z_{id}))$. SOM algorithm, defines a mapping from *C* to *A* where a neuron *c* is associated to its referent vector $w_c=(w_{c1}...w_{cj}....w_{cd})$ in $R^d$. At the end of the learning algorithm two neighbouring neurons on the map have close referent vector in the Euclidian space $R^d$. These SOM proprieties, (clustering, visualization and abstraction), are used in many complex tasks such as process analysis, machine perception, control, and communication [8].

The learning algorithm of SOM proceeds on 3 steps:

- *Initialization*: at time $t = 0$, we initialize a referent vectors $w_c$.
- *Competition*: At time $t > 0$, we assign an item *z* to the winner neuron using the assignment function $\phi$ defined as follows:

$$\phi(z) = \arg\min_{c \in C} \left\| z - w_c^{[t]} \right\|^2$$

- *Adaptation*: we update the referent vectors in the neighbouring of the winner neuron:

$$w_c^{[t]} = w_c^{[t-1]} + \varepsilon(t)K(\delta(r,c))( w_c^{[t-1]} - z)$$

$\varepsilon(t)$ : is a learning step at time *t*

In practice we run the algorithm until the minimization of a global error computed over the map or according to fixed number of learning cycles.

## 3. Feature selection

Clustering using SOM is an unsupervised learning technique used to discover group of structures in a dataset. Features selection or selection of variables for clustering is difficult because, unlike supervised learning, there are no class labels for the dataset and no obvious criteria to guide the search. The important issue of feature selection is to provide variables which give the "best" homogenous clustering. Therefore, the heuristic measure that we propose, consists of selecting the most important set of variables, which provide the "best" map *C* obtained with SOM algorithm.

### 3.1. Heuristic measure HI-SOM

The heuristic measure proposed for feature selection using SOM, is based on the work of Yacoub and Bennani [12]. The authors have proposed a heuristic measure called HVS for multi layer perceptron (MLP). HVS evaluates an importance of each variable for estimating a process. Thus, HVS uses both the connections of weights and the structure of the network. Therefore, an importance of a given neuron in input layer is estimated with respect to total partial contributions of MLP's outputs.

The main difference with HVS method is that the proposed HI-SOM measure is done for clustering. As known, clustering with SOM consists of projecting the samples from high-dimensional data set to a 2-dimensional topological map. Thus, each neuron in the map is activated or not by a subset of data. Hence, we adapt the proposed method by adding a new parameter, which represents the density of each neuron $c \in C$ denoted by $\gamma_c$ and defined as follows:

$$\gamma_c = Card(c) / N$$

Where $Card(c)$ represents the number of items assigned to the neuron *c*. The main idea of our proposed measure is to evaluate for each variable, its influence over the activation of each neuron in the map. For example, for an inactive neuron (empty neuron), the influence is null for all variables because there is no item (example)

assigned to this neuron. Thus, there are no important variables for this neuron.

Thus, the importance of variable $j \in \{1,2,3,...,d\}$ (feature of database) associated with each neuron $c \in C$ is defined as follows:

$$h_{jc} = \gamma_c \cdot \frac{w_{cj}}{\sum_{k=1}^{d} w_{ck}}$$

Therefore, the importance of variable $j \in \{1,2,3,...,d\}$ associated with the total clustering results provided by SOM is defined as follows:

$$\overline{h_j} = \sum_{c=1}^{N} h_{jc} \quad .....(1)$$

We notice that HI-SOM evaluates the distribution of samples in the neurons and the relationship of each feature with the others.

### 3.2. Model selection

As HVS, we propose to select features using "Backward" procedure based on HI-SOM measure (expression 1) . We delete the variables one after one in iterative way. After each variable deletion, learning is done again with the remainder of variables. Let $C_p$, $p$ = 1,…,*d-1* denote the maps provided at each step. The principal stages of the learning algorithm HI-SOM are:

1. $p$=1, $S_p$ = {1,2,...,*j*,...,*d*} (all variables)
2. Learning a map $C_p$ with selected variables $S$ using SOM algorithm defined above
3. Calculating $\overline{h_j}$, $1 \le j \le d$
4. Deleting the variable $j$ associated to minimum $\overline{h_j}$,
   $j = \arg\min_{j=1..d} \overline{h_j}$
   $S_{p+1}=S_p-\{j\}$
5. Evaluating the map's quality
6. Repeat 2 until the last variable

The feature selection consists of select the set of variables, which give the best quality of the map after SOM training.

The algorithm presented above generates a set of *d-1* maps (*d* is the number of variables) having less and less variables. Let note *TE(p)* the estimated topological error which represents data set topology representation accuracy. This topographic error measure presents percentage of data vectors, for which the first and second best match units (BMUs) are not adjacent units.

We note also *GDR(p)* as the global decrease error of learning defined as follows: $GDR(p) = \frac{E_{i(p)} - E_{f(p)}}{E_{i(p)}}$ where, $E_i(p)$ et $E_f(p)$ represent respectively the initial error and final error. *GDR* represents data representation accuracy using average quantization error between data vectors and their best match units (BMUs) on the map before and after training.

Two different approaches can be used for selecting the subset of variables:

- The first approach consists of choosing the map $C_{p^*}$, which obtains the least topological error

$$C_{p^*} = \arg\min_{S_p} TE(p)$$

And choose the subset $S_{p^*}$ of variables associated to $C_{p^*}$

If we obtain two subsets $S_{p'}$ and $S_{p''}$ associated respectively to two similar *TE(p'), TE(p")*; we select the one which has the high value of *GDR*.

- The second approach consists of using a statistical test (Fisher test) and search among all the maps $C_p$ those which are statistically close to $C_{p^*}$

## 4. Experiment results

### 4.1. Data

We have performed several experiments on three known problems from the UCI Repository of machine learning databases [2]. The chosen databases are: "Iris", "Glass" and "Cleve" (see Table1).

| Database | #data | #Variables | #classes |
|---|---|---|---|
| Iris | 150 | 4 | 3 |
| Glass | 214 | 9 | 6 |
| Cleve | 303 | 14 | 2 |

**Table1. The number of samples, variables and classes in the various data sets**

## 4.2. Clustering results

***"Iris" database***

The samples in this database are characterized with 4 variables. We apply HI-SOM heuristic for selecting the best set of variables allowing us to obtain the best map's quality.

At each step we calculate both: topological error (TE) and global decrease rate (GDR).

| Var | 1 | 2 | 3 | 4 | GDR(%)/TE |
|---|---|---|---|---|---|
| Step1 | 24.6 | 24.68 | 26.09 | 24.62 | 36.78/0.013 |
| Step2 | - | 31.90 | 35.36 | 32.72 | 42.51/0.033 |
| Step3 | - | - | 52.37 | 47.63 | **65/0.013** |

**Table 2. Importance of iris variables ($\overline{h}$) during backward procedure**

Table 2 represents the importance of selected variable in Step1 $S_1$={1,2,3,4}, Step2 $S_2$={2,3,4}, Step3 $S_3$={3,4}, respectively. The importance of variables is calculated by HI-SOM measure after training SOM with these sets.

We can observe that the best set of variables is {3: Petal length, 4: Petal Width} given the best quality for the map on both: TE and GDR.

***"Glass" database***

The samples in this database are characterized with 9 variables. After application of HI-SOM, we obtain the results illustrated in Figure 2.

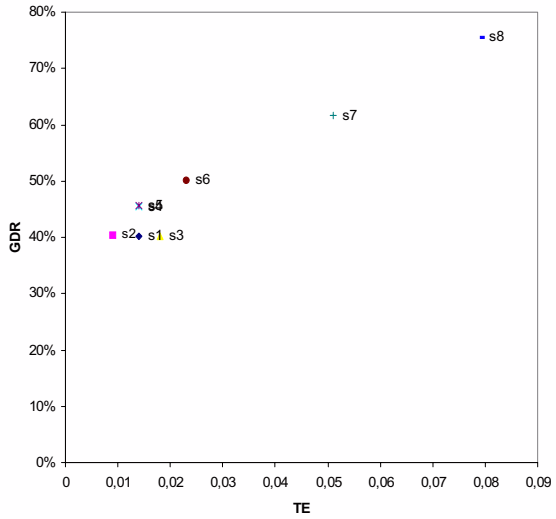



**Figure 2. Quality of the maps over different set of variables. (TE vs GDR).**

In Figure 2, HI-SOM provides the following set of variables:

$S_1$ = all variables;
$S_2$= {1,2,3,4,5,7,8,9}
$S_3$= {1,2,3,4,5,7,9}
$S_4$= {1,2,3,5,7,9}
$S_5$= {1,2,3,7,9}
$S_6$= {1,2,3,9}
$S_7$= {1,3,9}
$S_8$= {3,9}

The figure 2 shows that $S_2$ represents the best set of variables with TE= 0.09 and GDR = 40%. Thus, we can conclude that all variables are important except the variable "6" (Silicon) for clustering "Glass" database. Figure 3 shows also that the variable 6 has the lowest value for h in the initial clustering using all variables.

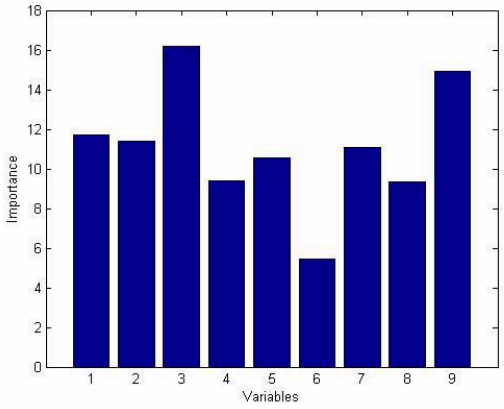



**Figure 3. Importance ($\overline{h}$) of variables for the first clustering of "Glass" using all variables**

***"Cleve" database***

The last dataset was *Cleve* dataset, which is Dr. Detrano's heart disease dataset. It was generated at the Cleveland Clinic modified to be a real mixed dataset. The dataset has 303 instances, each being described by 6 numeric and 8 categorical attributes. The instances were also classified into two classes, each class is either healthy (buff) or with heart-disease (sick). The application of our algorithm over this dataset gives variables {3, 4} as a best set of variables which provides the best map with TE= 0.019 and GDR = 43.18%.

To evaluate the quality of clustering, we adopt the approach of comparing the results with

clustering accuracy. This a common approach in the general area of data clustering.

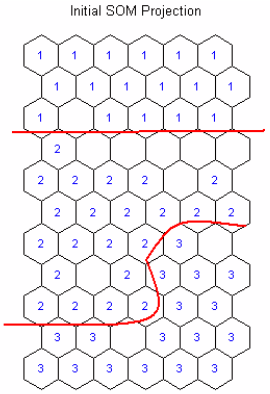
Subsequently, the overall accuracy is given for each dataset in Table.3 (Iris, Glass, Cleve).

| Database | Initial SOM (#variables) | After HI-SOM (#variables) |
|---|---|---|
| Iris | 96% (4) | 96% (2) |
| Glass | 74% (9) | 74,15% (8) |
| Cleve | 70,3% (14) | 74,91% (2) |

**Table 3. Accuracy comparison**

Thus, we can see in Table3 that the accuracy is improved after using HI-SOM, because the map is constructed with a best set of variables from databases. This improvement is clearly showed with Cleve dataset than the others. This can be explained by the initial dimension of datasets which is more important than "Iris" and "Glass". However, we can see for "Iris" in figure 4 the difference between the map provided with all variables and the map after applying HI-SOM.

(a)




(b)

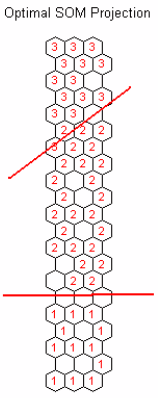



**Figure 4. Projection of Iris labels after applying a majority vote rule. (a) using standard SOM with all variables, (b) using HI-SOM.**

We know that in "Iris" database there is 3 classes of 50 instances for each one. Each class refers to a type of iris plant. The figure 4.a shows that the class 1 is linearly separable from the two others (2 and 3). The class 2 and 3 are not linearly separable from each other. However after applying HI-SOM, the display of the labels (1,2, 3) are generaly separated each one from the others (Figure 4.b). This clearly shows that using SOM with selected feature provides a good separation between classes.

## 5. Conclusion

We have proposed a new heuristic measure for optimizing the input data for self organizing map algorithm. This heuristic, has been proposed to select features for clustering problems. Using this measure, we have determined the importance of each variable for the obtained classes and then optimized the input data of SOM paradigm by selecting the best set of variables which allows obtaining the best map's quality. In the future work, we propose to directions: in one hand, to use HI-SOM and adapt it to topological map for categorical and mixed data and in the other hand, to develop a local optimization of the map using HI-SOM measure. In fact with this heuristic we can select an optimal number of features for each neuron in the map according to the projection of data.

## 6. References

[1] Bennani,Y . " Systèmes d'apprentissage connexionnistes, sélection de variables". Revue d'intelligence artificielle. Volume 15 – n° 3-4/2001.

[2] Blake, C.L and Merz, C.MJ. "UCI repository of machine learning databases". Technical report. University of California, Department of information and Computer science, Irvine, CA, 1998. available at: ftp://ftp.ics.uci.edu/pub/machine-learning-databases/

[3] Brusco, M.J, and Cradit, J.D. "A variable-selection heuristic for k-means clustering". Psychometrika – Vol 66 N°2, 2001, 249-270.

[4] Dash, M and Lui, H. "Feature selection for clustering". Pacific-Asia Conference on Knowledge Discovery and Data Mining. 2000.

[5] Hsu, A, Alahakoon, D, Halgamuge, S,.K, and Srinivasan, B, "Automatic clustering and rule extraction using a dynamic SOM tree," Proceedings of the 6th International Conference

on Automation, Robotics, Control and Vision, Singapore, 2000.
[6] Kittler, J. “Feature selection and extraction”. In Handbook of Pattern Recognition and Image Processing, Eds. T. Y. Young, K.S. Fu, Academic Press, pp. 59-83, 1986.
[7] Koggalage, R, Halgamuge, S and Hsu, A.L.C. “Unsupervised class discovery and feature selection using an improved hierarchical dynamic Self-Organizing Map”. Neural Information Processing - Letters and Reviews Vol. 3, No. 1, April 2004.
[8] Kohonen, T. “Self-Organizing Maps”. Series in Information Sciences, Vol. 30, Springer, Berlin, Heidelberg, New York, 1995, 1997, 2001. Third Extended Edition, 501 pages. ISBN 3-540-67921-9, ISSN 0720-678X.
[9] Law, M.H.C, Figueiredo, M.A.T and Jain, A.K. “Simultaneous feature selection and clustering”.
http://www.lx.it.pt/~mtf/IEEE_TPAMI_2004_Law_Figueiredo_Jain.pdf.
[10] Ribert, A Ennaji, A and Lecourtier, Y. “An incremental hierarchical clustering”. Vision interface 99, Trois rivières, Canada, 1999.
[11] Vesanto, J, Himberg,J, Alhoniemi, E, and Parhankangas, J. “SOM ToolboxTeam”. Helsinki University of Technology. P.O.Box 5400, FIN-02015 HUT, FINLAND.
http://www.cis.hut.fi/projects/somtoolbox/.
[12] Yacoub, Y and Bennani, Y. Feature selection and architecture optimization in connectionist systems. International journal of Neural Systems , Vol 10, 5, (2000), 379-395.
[13] S. Guérif and Y. Bennani. Selection of Clusters Number and Features Subset During a Two-Levels Clustering Task. in Artificial Intelligence and Soft Computing. August 28-30, 2006, Palma de Mallorca, Spain
[14] Volker.R and Tilman.L. FeatureSelection in Clustering Problems. Advances in Neural Information Processing Systems 16, Edit. Sebastian Thrun and Lawrence Saul and Bernhard. MIT Press. Cambridge, MA. 2003.
[15] Cheeseman. P, Kelly. J, Self. M, Stutz. J, Taylor. W and Freeman. D. Autoclass: A Bayesian classification system. Proceeding of the Fifth International Workshop on Machine learning . pp 54-64. Ann Abor, MI: Morgan Kaufmann, 1988.
[16] Fisher. D. Iterative optimization and simplification of hierarchical clustering. Journal of Artificial Intelligence Research, 4, 147-179.