Dimensionality Reduction Through Unsupervised Features Selection

Sébastien Guérif, Younès Bennani
University of Paris 13, LIPN / CNRS UMR 7030
99, avenue JB. Clement - F-93430 Villetaneuse, France
{sebastien.guerif, younes.bennani}@lipn.univ-paris13.fr

Abstract

As the storage technologies evolve, the amount of available data explodes in both dimensions: samples number and input space dimension. Therefore, one needs dimension reduction techniques to explore and to analyse his huge data sets. Many features selection approaches have been proposed for the supervised learning context, but only few techniques are available to address this issue in the unsupervised learning context. Actually, the problem of unsupervised feature selection becomes more difficult as the samples points' labels disappear. Thus, most of the methods proposed rely on feature correlations and only pairs of variables are considered. In this paper, we extend the wkmeans algorithm proposed by Huang to the self-organizing maps (SOM) framework and we propose a feature selection approach which relies on the weighting coefficients learned during the optimization process. This SOM-based approach addresses the difficult issue of unsupervised feature selection and is ready to handle high dimensional data sets.

1 Introduction

In a recent study [9] about the most often used data mining or analytic methods used in the year 2006, clustering is placed at the second rank just behind the decision trees and association rules with near 40 % of the voters. Although the success of the clustering approach for exploratory analysis is uncontested, the methods have to be adapted to deal with more and more data. Actually, as the storage technologies evolve, the amount of available data "explodes" in both dimensions: sample size and input space dimension. Now, in machine learning the number of necessary sample points grows exponentially with the dimension of the feature space; this problem is known as the *curse of dimensionality*. Therefore, one needs techniques to reduce the dimension of the sample points description and should use either features extraction, features selection or a combination of the both.

The features extraction approaches build new features using the original variables while the features selection techniques select the most relevant dimensions. Although the former generally achieves to higher accuracy classifiers in supervised learning context, the latter leads to a more understandable description of the data samples. In this paper, our purpose is to furnish the user with a minimum effort solution to get an insight into the hidden knowledge of his data and to assess the relevancy of further analysis. Therefore, the approach proposed includes both a feature selection approach and a powerful visualisation technique, namely the self-organizing maps (SOM).

The remainder of this paper is organized as follows: first, section 2 presents briefly some related works, then the section 3 presents the k-means and the self-organizing maps algorithms which are extended in the section 4. The weighting based feature selection approach proposed is developped in the section 5 and the experimental results are shown in the section 6. Perspectives and further research are presented as a conclusion.

2 Related Work

Most of the unsupervised dimension reduction approach are feature extraction methods such the Multidimensional Scaling (MDS), the Isometric Mapping (Isomap) [14] or the Locally Linear Embedded (LLE) [13]. But these techniques are computationnally expensive because they require an estimation of the geodesic distance matrix between sample points and the computation of its inverse. Thus, whereas some techniques of incremental computation have been recently developped, they are not suitable since the dataset are large.

Although feature selection has been extensively studied in the context of supervised learning, this field is relatively new in the unsupervised learning. A broad part of the methods which proposed to achieve feature selection in the unsupervised context try to eliminate the redundancy among a feature subset and thus they rely either on correlation or an estimation mutual information [6, 12, 15]. Actually, P.

Mitra and al. uses a similarity measure that corresponds to the lowest eigenvalue of correlation matrix between two features [12], J. Vesanto and al. proposed to visually detect correlation using a SOM-based approach [15] and S. Guérif and al. used a similar idea and integrated a weighting mechanism in the SOM training algorithm to reduce the redundancy side effects [6]. Some others techniques try to approximate an ultra-metric in an euclidian space [11] or to preserve the set of the k-nearest neighbors. More recently, some approaches have been proposed to adress the difficult issue of irrelevant features elimination in the unsupervised learning context [1, 5]; these approaches use quality measures of partition such the Davies-Bouldin index [2, 5], the Wemmert and Gancarski index or the entropy [1].

3 K-Means and the Self-Organizing Maps

3.1 Notations

Let $X = \{x_i \in \mathbb{R}^n : i = 1, \ldots, N\}$ be a set sample points; N et n respectively stands for the sample size and the input space dimension. A partition $P = \{P_j : j = 1, \ldots, K\}$ of X can be represented by a partition matrix $U = (u_{ij})$ where the indexes $i = 1, \ldots, N$ and $j = 1, \ldots, K$ refer respectively to one sample point $x_i \in X$ and to one cluster $P_j \in P$; thus, $u_{ij} = 1$ means that the sample point $x_i \in X$ belongs to the cluster $P_j \in P$ and for all $i = 1, \ldots, N$ we have $\sum_j u_{ij} = 1$.

3.2 K-Means algorithm

K-means [4] is one of the most widely used clustering algorithm. Each cluster is represented by a prototype $z_j \in \mathbb{R}^n$ that belongs to the input space. K-means algorithm optimizes the following cost function:

$$R_{KM}(U, Z) = \sum_i \sum_j u_{ij} \times d^2(x_i, z_j) \tag{1}$$

where U is a partition matrix, Z is the matrix whose rows are the clusters'prototypes and $d : \mathbb{R}^n \times \mathbb{R}^n \rightarrow \mathbb{R}_+$ is the dissimilarity measure used to cluster the sample points. A common choice for the dissimilarity measure when one uses the k-means algorithm is the euclidian distance which is defined as follows:

$$d^2(x_i, z_j) = \sum_k (x_{ik} - z_{jk})^2 \tag{2}$$

where x_i and z_j are respectively a sample point from X and the prototype of the cluster $P_j \in P$. The loss function given by (1) can be optimized by selecting randomly the initial prototypes and then iterating the two following steps until convergence:

1. Affectation optimization: the clusters'prototypes $\hat{Z}$ are fixed and the cost function $R_{KM}(U, \hat{Z})$ is optimized by assigning the cluster with the nearest prototypes to each sample point,

$$u_{ij} = \begin{cases} 1, & \text{if } j = \underset{P_l \in P}{argmin}\, d^2(x_i, \hat{z}_l), \\ 0, & \text{otherwise.} \end{cases} \tag{3}$$

2. Prototypes optimization: the partition matrix $\hat{U}$ is fixed and the cost function function $R_{KM}(\hat{U}, Z)$ is optimized by updating each cluster prototype with the mean of the sample points assigned.

$$z_j = \frac{\sum_i \hat{u}_{ij}\, x_i}{\sum_i \hat{u}_{ij}} \tag{4}$$

Ones used to reproach this algorithm for its main weakness: the initial prototypes determine the solution and the algorithm often converges to a local minimum.

3.3 Self-Organizing Maps

The Self-Organizing Maps (SOM) was introduced in the early 80's by Prof. Teuvo Kohonen as a multidimensional data visualization method [10]. This approach can be seen as an enhancement of the k-means algorithm introduced above where the clusters'prototypes are subjected to a neighborhood constraints that preserve the topological ordering of the input space. Therefore, the clusters'centers, also refers as units or neurons, are organized according a lattice to form a map. The length of the shortest path between two units, expressed as the number of edges that separates them on the map, defines a distance δ in the projection space. The topological ordering preservation means that neighbors sample points in the input space have to be assigned to the same unit or to neighbors units. This can be achieved by introducing the neighborhood constraint in the k-means cost function (1) in the following way:

$$R_{SOM}(U, Z) = \sum_i \sum_j u_{ij} \times \left[\sum_l h_{jl} \times d^2(x_i, z_l) \right] \tag{5}$$

where h_{jl} is the value of the neighborhood function between the units j and l. A gaussian kernel parameterized by λ is usually used as neighborhood function:

$$h_{jl} = exp\left(- \frac{\delta^2(j, l)}{2\, \lambda^2} \right) \tag{6}$$

The SOM cost function (5) is optimized either by a gradient descent approach, or by a procedure similar to the one

described above for the k-means cost function. The second step has to be modified to take into account the neighborhood function, and the prototypes are updated using the weighted centroid of the sample points assigned to one unit or its neighbors. One should be aware that the value of the gaussian kernel parameter λ have to decrease over the time and that a too small value at the beginning can avoid the self-organization process.

4 Weighting feature during clustering

4.1 The ω-k-means algorithm

Huang and al. [8] proposed to introduce a weighting coefficient in the k-means cost function by replacing the euclidian distance by the weighted euclidian distance defined as:

$$d_\omega^2(x_i, z_j) = \sum_k \omega_k^\beta \times (x_{ik} - z_{jk})^2 \quad (7)$$

with $\omega_k \geq 0$ and $\sum_k \omega_k = 1$. Therefore, they modified the k-means cost function as follows:

$$R_{\omega KM}(U, Z, W) = \sum_i \sum_j u_{ij} \times d_\omega^2(x_i, z_j) \quad (8)$$

where W is the column matrix of the weighting coefficients used in the computation of d_ω. The k-means optimization procedure presented in the previous section can be used to optimize the cost function above (8); the weighting coefficients $\hat{W}$ are fixed during the two first steps and according to the theorem given in [8], the following third additional step optimizes (8) against W:

3. Weights optimization: the partition matrix $\hat{U}$ and the clusters'prototypes $\hat{Z}$ are fixed and it is shown in [8] that the cost function $R(\hat{U}, \hat{Z}, W)$ attains its minimum for the following weighting coefficients values:

$$\omega_k = \begin{cases} 0, & \text{if } D_k = 0, \\ \left(\sum_t \left[\frac{D_k}{D_t} \right]^{\frac{1}{\beta - 1}} \right)^{-1}, & \text{otherwise.} \end{cases} \quad (9)$$

$$\text{with } D_k = \sum_i \sum_j \hat{u}_{ij} \times (x_{ik}, \hat{z}_{jk})^2 \quad (10)$$

4.2 The ω-SOM algorithm

The ω-k-means algorithm presented above can be extend to the SOM framework by introducing neighborhood contraints between prototypes to preserve the input space topological ordering. Therefore, the ω-k-means and the SOM cost functions can be combined as follows:

$$R_{\omega SOM}(U, Z, W) = \sum_i \sum_j u_{ij} \times \left[\sum_l h_{jl}\, d_\omega^2(x_i, z_l) \right] \quad (11)$$

The optimization of the ω-SOM cost function (11) is achieved using the same algorithm as for the ω-k-means one (8). Whereas the expression (10) of the D_k terms used in (9) have to be rewritten as follows:

$$D_k^{(SOM)} = \sum_i \sum_j \hat{u}_{ij} \times \left[\sum_l h_{jl}\, (x_{ik} - z_{lk})^2 \right] \quad (12)$$

the proof of the theorem proposed in [8] remains valid. In the high level algorithm 1 that summarizes the ω-SOM optimization procedure, t and T_{max} refers respectively to the current iteration and to the number of training epochs.

Algorithm 1 ω-SOM algorithm

Initialize W and Z at random.
for $t = 1, \ldots, T_{max}$ **do**
  Optimize $R_{\omega SOM}(U, \hat{Z}, \hat{W})$: each sample point is assigned to its best matching unit according the d_ω distance measure,
  Optimize $R_{\omega SOM}(\hat{U}, Z, \hat{W})$: clusters'prototypes are updated using the weighted centroid of the sample points that belong neighborhood units,
  Optimize $R_{\omega SOM}(\hat{U}, \hat{Z}, W)$: weighting coefficients are updated according the theorem proposed in [8] extended to the SOM framework.
end for

5 From Weighting Feature to Feature Selection

It is often argued that the feature extraction methods lead to more accurate classifiers, but in the context of clustering feature selection approaches should be preferred; actually, the remaining dimensions are directly understandable by the user and do not require any additional interpretation effort as the features extracted need. A feature selection procedure is composed from three following essential elements: a pertinence measure, a search procedure and a stop criterion.

5.1 Pertinence measure

In the supervised learning context, the pertinence of a feature subset is often evaluated according a model performance criterion that depends on the task it has been designed for: regression or classification. In the unsupervised learning context, define a pertinence measure become more difficult because there is neither a value to predict nor a correct class to assign to each samples points. Anyway, the weighting coefficients learned by the ω-SOM algorithm give the relative importance of each dimension: the bigger

is ω_k, the more the k-th dimension contributes to the clustering result. Therefore, the features'weights provide us a reliable pertinence measure.

5.2 Search procedure and Stop criterion

To find an optimal solution involves an exhaustive search over the $2^n - 1$ possible feature subsets. Although efficient algorithm as *Branch and Bound* have been proposed, they requires the monotonicity of the evaluation criterion which is usually difficult to insure. Therefore, the exhaustive approach is infeasible since n is large and we have to adopt an suboptimal approach. Anyway, the pertinence measure proposed in the previous paragraph defines an ordering relation between features and not between features subsets. Therefore, the criterion proposed suggests the use of a nested subset approach such forward selection or backward elimination. The former starts with the empty set and the features are progressively added according their decreasing relevance, whereas the latter begins with the whole feature set and removes the less interesting features. It is often argued that the forward selection is computationnally more efficient, but the backward elimination has been preferred in this paper because it takes into account the mutual pertinence of features; two dimensions can be interesting when they are considered together whereas each is individually uninteresting.

Anyway, a correct feature selection procedure requires an update of the pertinence measure after the feature removing or adding, but this is not feasible since we want to address the problem of feature selection with high dimensional datasets. Therefore, we use a criterion based on the following assumption: the lowest weighting coefficients which share the same order of magnitude corresponds to uninteresting dimensions and a significant change of order should indicates that the other features might be selected. To detect this change, weighting coefficients are sorted and we compute the ratio between adjacent weights. Assuming these ratio are normally distributed, we can considered that a significant change occurs when the ratio value exceeds a trueshold $\theta = \mu + t_\alpha \, \sigma$ where μ and σ are respectively the mean and the standard deviation of the ratio values. We set $t_\alpha = 1.64$ which corresponds to $\alpha = 5\%$ in the unilateral case.

5.3 Feature selection algorithm

The outlines of the feature selection approach proposed are summarized by the following high-level algorithm:

Algorithm 2 Feature selection algorithm

Standardize the data (zero mean and unit variance).
Train a SOM using the ω-SOM algorithm
Sort the weighting coefficients W in ascending ordering
Compute ratios $q(i) = \frac{\omega(i+1)}{\omega(i)}$
Search the first $q(i) > \mu + t_\alpha \, \sigma$
Select all features with a weight greater than $\omega(i)$

6 Experimental results

6.1 Datasets

We used several datasets with different size and complexity to evaluate our approach and the results from two datasets are presented in this paper: the first dataset is from the UCI repository [3] and the second one was proposed during the NIPS'2003 Feature Selection Challenge [7]:

- The *waveform* dataset is composed of 5000 sample points from three classes. Each classes has been generated from a combination of 2 of 3 *base* waves and a gaussian noise has been added in each dimension. The original dataset was in dimension 21 but 19 additional normally distributed noisy dimensions has been added.
- The *madelon* dataset is a 2 classes problem originally proposed in the NIPS'2003 feature selection challenge [7]. The samples points are situated on the vertices of a five dimensional hypercubes, but 15 redundant features and 480 probes has been added. The *probes* dimensions are distributed according the same distribution that the interesting features but they are independant from the labels assigned to each vertices by random. The original dataset was splitted into 3 parts (training, validation and testing subsets) and we use only the 2600 sample points from the training and the validation subsets because labels from the testing subset were not available.

6.2 Evaluation methodology

A ten-folds cross-validation approach was used to evaluate the performance of our approach: 90 per cents of the samples points were used for training and the remaining 10 per cents were used for the performance evaluation. The class labels of the sample points were available, thus the accuracy of a k-nearest neighbors classifiers could be used to evaluate the relevance of the feature subset selected. Then, the significance of the accuracy improvement was verified by comparing the results with those obtained 10 different random permutations of the weighting coefficients. Each experiments was repeated for the value from 2 to 10 of the parameters β.

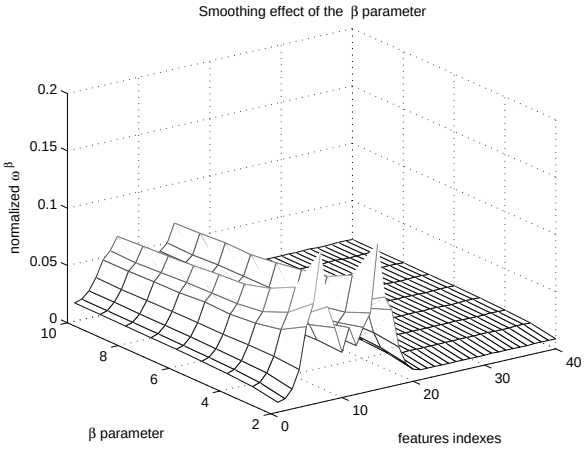

Figure 1. Smoothing effect of the β parameter (median weights obtained for the *waveform* dataset): the weighting coefficients ω_j^β have been normalized such that $\sum \omega_j^\beta = 1$. One can observed that as the β values increases, the weights become more smoothed.

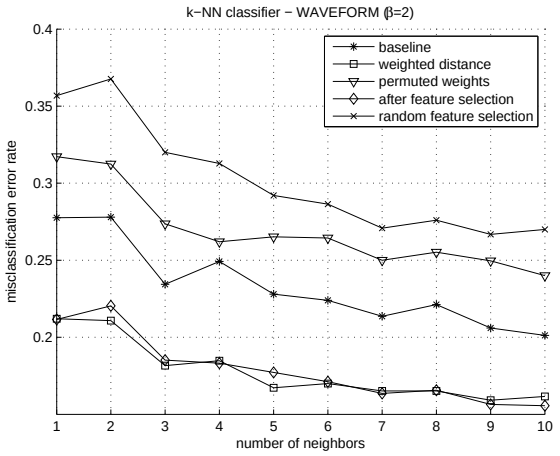

Figure 2. k-nearest neighbors classifiers accuracies using 90 per cents of the *waveform* dataset as training set with $\beta = 2$.

6.2.1 Selected subset relevance

The figure 2, 3 and 4 show respectively the k-nearest neighbors classifiers accuracies one the *waveform* for the values 2, 5, and 10 of the parameter β. The worst results are obtained for a random feature selection and the best one appears when the feature selection approach proposed is used. It should be noticed that as the value of β increases, the gap between the baseline and the distance with permuted weights decreases. This phenomenon is due to the smoothing effect of the parameter β which is highlighted by the figure 1. The same phenomenon explains why the gap between the line corresponding to the non permuted weighted distance and the baseline decrease as β increases. Anyway, the feature selection approach proposed, which does not used the information about class labels, leads to an accuracy improvement of more than 5 points regardless of the parameter β values.

The behavior of the weighted distances (with or without weights permutation) is confirmed by the figure 5, 6 and 7. One should notice that the approach proposed leads to an accuracy improvement of at least 15 points with the *madelon* dataset.

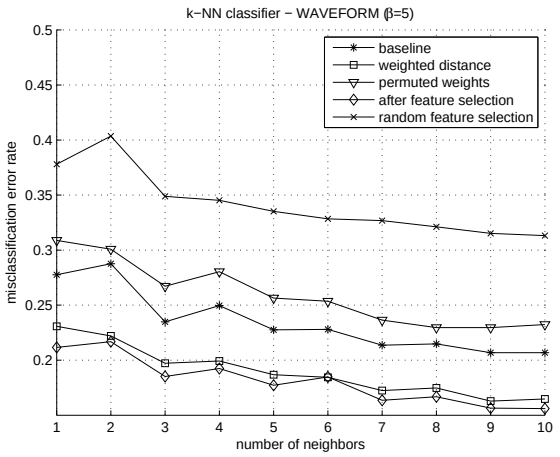

Figure 3. k-nearest neighbors classifiers accuracies using 90 per cents of the *waveform* dataset as training set with $\beta = 5$.

6.2.2 Selected subset stability

For each runs of the feature selection method proposed, except one, the features 3 to 19 of the *waveform* dataset were selected. The only run which leads to a different result se-

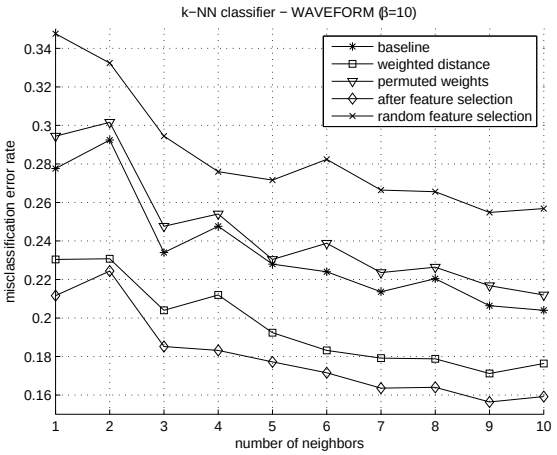

Figure 4. k-nearest neighbors classifiers accuracies using 90 per cents of the *waveform* dataset as training set with $\beta = 10$.

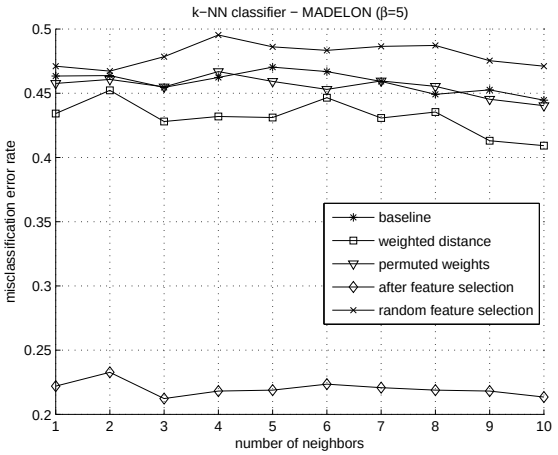

Figure 6. k-nearest neighbors classifiers accuracies using 90 per cents of the *madelon* dataset as training set with $\beta = 5$.

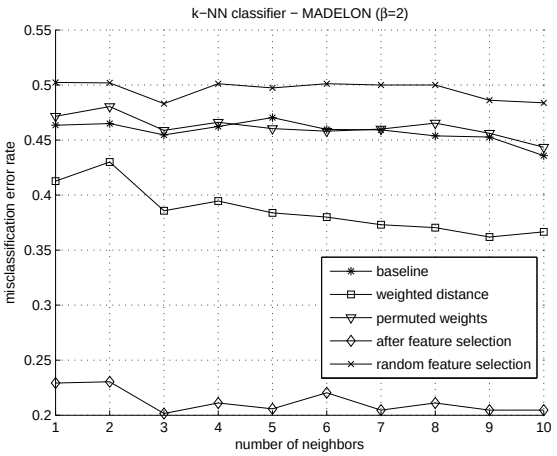

Figure 5. k-nearest neighbors classifiers accuracies using 90 per cents of the *madelon* dataset as training set with $\beta = 2$.

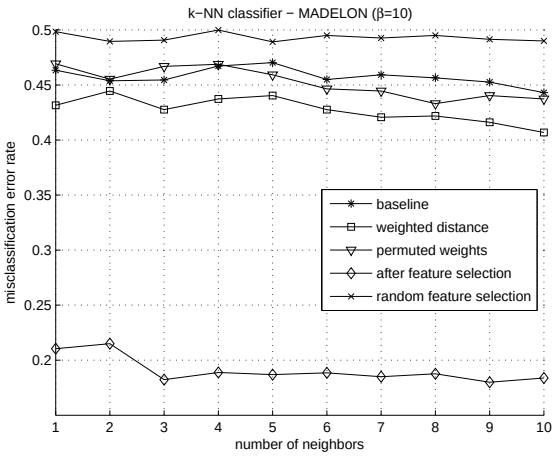

Figure 7. k-nearest neighbors classifiers accuracies using 90 per cents of the *madelon* dataset as training set with $\beta = 10$.

lected features 4 to 18. The stability of our algorithm was also observed when it was applied to the *madelon* dataset with reasonable values of the parameter β. Actually, the features 29, 65, 106, 129, 154, 242, 282, 319, 337, 339, 434, 443, 452, 454, 473, 476 and 494 from the *madelon* dataset were selected for more than half of the execution for each β values. Anyway, the little decreasing of stability observed is once again related to the smoothing effect of the β parameters which affects the cutting value of the two adjacent weights ratio.

6.2.3 Discovery of true classes

After the feature selection process has been achieved, one would like to get an insight into the structure of his data. The two-levels clustering approach proposed in [16] can be used for this purpose: a self-organizing map is trained using the reduced features subset and the k-means algorithm is applied to cluster the maps units. The number of clusters can be determined using the Davies-Bouldin index [2] which is defined as follows:

$$I_{DB} = \frac{1}{K}\sum_{j=1}^{K} \underset{j\neq l}{max} \left\{ \frac{\sum_i u_{ij} \ \|x_i - z_j\|^2 + u_{il} \ \|x_i - z_l\|^2}{\|z_k - z_l\|^2} \right\} \tag{13}$$

where K is the number of clusters. This clustering procedure aims to find internally compact spherical clusters which are widely separated.

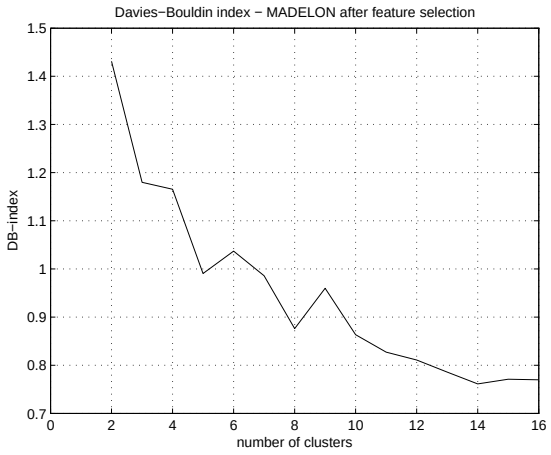

Figure 8. Davies-Bouldin index for different number of clusters when the SOM of the reduced *madelon* dataset is clustered. The minimum 0.76 is reached for $K = 14$.

The figure 8 shows the Davies-Bouldin index values for different segmentation of the SOM trained with the selected features subset from the *madelon* dataset. According to this criterion, the best segmentation of the map is obtained for 14 clusters and is presented by the figure 9. The contingency table 1 shows us the repartition of the true classes in the 14 clusters discovered. One should notices that the accuracy of this classification which does not use the class labels information is better than the baseline of the k-nearest neighbors classifiers, and is not really worse than the performance obtained by the supervised classifier afeter feature selection.

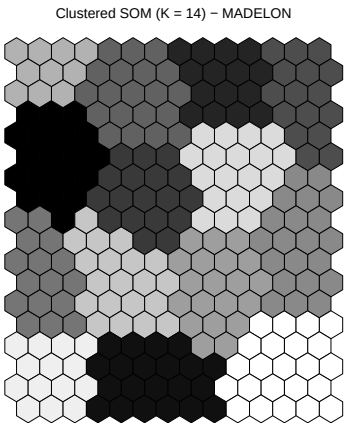

Figure 9. The SOM of the reduced *madelon* dataset is segmented in 14 clusters according the Davies-Bouldin index.

The figure 10 shows the Davies-Bouldin index values for different segmentation of the SOM trained with the selected features subset from the *waveform* dataset. According to this criterion, the best segmentation of the map is obtained for 6 clusters and is presented by the figure 11. The contingency table 2 shows us that the 6 clusters discovered correspond either to the real classes or to their important two by two overlapping; this is confirmed by a visual inspection of the relative position of clusters on the map.

7 Conclusion

In this paper, we propose a feature selection approach for unsupervised learning. It relies on the computation of weighting coefficients using a SOM based clustering algorithm which is more robust than the k-means algorithm and provides us stable weights. On the one hand, these weighting coefficients provides us a pertinence measure which considers the ability of a feature to structure the dataset and which takes into account the mutual pertinence. On the other hand, they permit a progressive evaluation of the features relevance and they avoid the definitive elimination at

Table 1. Composition of the 14 clusters: -1 and +1 corresponds to the real *madelon* classes and by assigning the majority class to each cluster, the misclassification rate is 0.31.

	-1	+1	Purety
Cluster 1	160	47	77.29 %
Cluster 2	112	113	50.22 %
Cluster 3	105	65	61.76 %
Cluster 4	72	48	60.00 %
Cluster 5	51	155	75.24 %
Cluster 6	49	107	68.59 %
Cluster 7	96	55	63.58 %
Cluster 8	87	169	66.02 %
Cluster 9	117	21	84.78 %
Cluster 10	66	138	67.65 %
Cluster 11	10	139	93.29 %
Cluster 12	163	22	88.11 %
Cluster 13	64	104	61.90 %
Cluster 14	148	117	55.85 %

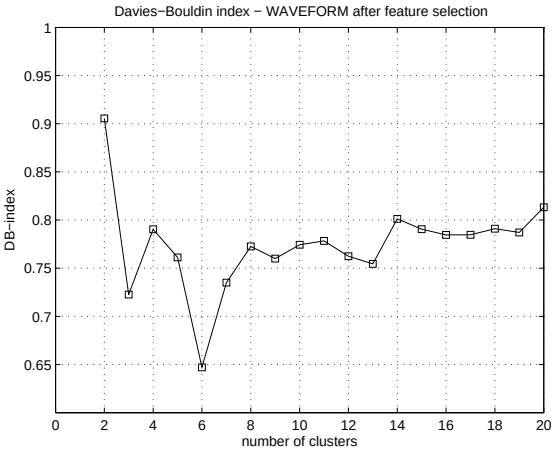

Figure 10. Davies-Bouldin index for different number of clusters when the SOM of the reduced *waveform* dataset is clustered. The minimum 0.65 is reached for $K = 6$.

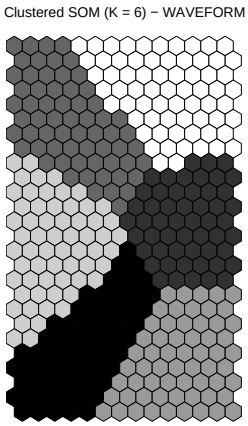

Figure 11. The SOM of the reduced *waveform* dataset is segmented in 6 clusters according the Davies-Bouldin index.

Table 2. Composition of the 6 clusters: 1, 2 and 3 corresponds to the real *waveform* classes and by assigning the majority class to each cluster, the misclassification rate is 0.30.

	1	2	3	Purety
Cluster 1	5	0	545	99.09 %
Cluster 2	602	2	3	99.18 %
Cluster 3	6	622	1	98.89 %
Cluster 4	577	615	0	51.59 %
Cluster 5	0	414	444	51.75 %
Cluster 6	502	0	662	56.87 %

the beginning of a backward procedure with a possibly incorrect pertinence measure such as in [5]. Next, the method proposed furnishes us with an efficient way to deal with high dimensional dataset. Future works includes an evaluation of the techniques proposed with tiny dataset in very high dimensional input space which are commonly used in bioinformatic or in spectrometry. In the latter case, the dimension are highly correlated and our method needs a significant enhancement to eliminate the redundancy by estimating the mutual information between features with the largest weights for instance.

References

[1] A. Blansché. *Classification non supervisée avec pondération d'attributs par des méthodes évolutionnaires*. PhD thesis, Universit Louis Pasteur - Strasbourg I, September 2006.

[2] D. L. Davies and D. W. Bouldin. A cluster separation measure. *IEEE Transactions on Pattern Analysis and Machine Intelligence, PAMI*, 1(2):224–227, 1979.

[3] C. B. D.J. Newman, S. Hettich and C. Merz. UCI repository of machine learning databases, 1998.

[4] R. O. Duda, P. E. Hart, and D. G. Stork. *Pattern Classification (2nd Edition)*. Wiley-Interscience, 2001.

[5] S. Guérif and Y. Bennani. Selection of clusters number and features subset during a two-levels clustering task. In *Proceedings of the 10th IASTED International Conference Artificial intelligence and Soft Computing 2006*, pages 28–33, Aug. 2006.

[6] S. Guérif, Y. Bennani, and E. Janvier. μ-som : Weighting features during clustering. In *Proceedings of the 5th Workshop On Self-Organizing Maps (WSOM'05)*, pages 397–404, Sep 2005.

[7] I. Guyon, S. Gunn, M. Nikravesh, and L. Zadeh. *Feature Extraction, Foundations and Applications, Editors*. Series Studies in Fuzziness and Soft Computing, Physica-Verlag. Springer, 2006, to appear.

[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):657–668, 2005.

[9] KDnuggets. 2006 kdnuggets poll on data mining/analytic techniques, http://www.kdnuggets.com/polls/2006/data_mining_methods.htm, April 2006.

[10] T. Kohonen. *Self-Organizing Maps*, volume 30 of *Springer Series in Information Sciences*. Springer, Berlin, Heidelberg, New York, third extended edition edition, 1995,1997,2001.

[11] V. Makarenkov and P. Legendre. Optimal Variable Weighting for Ultrametric and Additive Trees and K-means Partitioning: Methods and Software. *Journal of Classification*, 18(2):245–271, February 2001.

[12] P. Mitra, C. Murthy, and S. Pal. Unsupervised Feature Selection Using Feature Similarity. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 24(4), 2002.

[13] S. T. Roweis and L. K. Saul. Nonlinear Dimensionality Reduction by Local Linear Embedding. *Science*, 290:2323–2326, December 2000.

[14] J. B. Tanenbaum, V. de Silva, and J. C. Langford. A Global Geometric Framework for Nonlinear Dimensionality Reduction. *Science*, 290:2319–2323, December 2000.

[15] J. Vesanto and J. Ahola. Hunting for Correlations in Data Using the Self-Organizing Map. In H. Bothe, E. Oja, E. Massad, and C. Haefke, editors, *Proceeding of the International ICSC Congress on Computational Intelligence Methods and Applications (CIMA '99)*, pages 279–285. ICSC Academic Press, 1999.

[16] J. Vesanto and E. Alhoniemi. Clustering of the self-organizing map. *IEEE Transactions on Neural Networks*, 11(3):586–600, 2000.