k-EVCLUS: Clustering Large Dissimilarity Data in the Belief Function Framework*

Orakanya Kanjanatarakul[1], Songsak Sriboonchitta[2] and Thierry Denoeux[3]

[1] Faculty of Management Sciences,
Chiang Mai Rajabhat University, Thailand
[2] Faculty of Economics, Chiang Mai University, Thailand
[3] Sorbonne Universités, Université de Technologie de Compiègne, CNRS,
UMR 7253 Heudiasyc, France
orakanyaa@gmail.com, songsakecon@gmail.com, tdenoeux@utc.fr

Abstract. In evidential clustering, the membership of objects to clusters is considered to be uncertain and is represented by mass functions, forming a credal partition. The EVCLUS algorithm constructs a credal partition in such a way that larger dissimilarities between objects correspond to higher degrees of conflict between the associated mass functions. In this paper, we propose to replace the gradient-based optimization procedure in the original EVCLUS algorithm by a much faster iterative row-wise quadratic programming method. We also show that EVCLUS can be provided with only a random sample of the dissimilarities, reducing the time and space complexity from quadratic to linear. These improvements make EVCLUS suitable to cluster large dissimilarity datasets.

Keywords: Evidential clustering, Dempster-Shafer theory, evidence theory, unsupervised learning.

1 Introduction

Evidential clustering extends both hard and fuzzy clustering by modeling cluster-membership uncertainty using Dempster-Shafer mass functions. The collection of mass functions for n objects is called a *credal partition*. The first evidential clustering algorithm, called EVCLUS, was introduced in [4]. This algorithm constructs a credal partition from a dissimilarity matrix, in such a way that more dissimilar objects are assigned mass functions with greater degrees of conflict. This method was shown to perform as well as or better than other relational clustering algorithms on a variety of datasets, even when the dissimilarities are not Euclidean distances [4]. However, as other relational clustering algorithms, EVCLUS requires to store the whole dissimilarity matrix; the space complexity is thus $O(n^2)$, where n is the number of objects, which precludes application to

* This research was supported by the Labex MS2T, which was funded by the French Government, through the program "Investments for the future" by the National Agency for Research (reference ANR-11-IDEX-0004-02). It was also supported by the Center of Excellence in Econometrics at Chiang Mai University.

datasets containing more than a few thousand objects. Also, each iteration of the gradient-based optimization algorithm used in [4] requires $O(f^3n^2)$ operations, where f is the number of focal sets of the mass functions, i.e., the number of subsets of clusters being considered. This computational complexity of EVCLUS further restricts its use to relatively small datasets.

After EVCLUS, other evidential clustering algorithms were introduced. The Evidential c-means algorithm (ECM) [7] is an evidential version of the hard and fuzzy c-means; it is only applicable to attribute data. A version of ECM for dissimilarity data (Relational Evidential c-means, RECM) was later proposed in [8]. This algorithm is faster than EVCLUS, but it can fail to converge when the dissimilarities are not Euclidean distances. In [11], Zhou et al. introduced another variant of ECM, called the Median Evidential c-means (MECM), which is an evidential counterpart to the median c-means and median fuzzy c-means algorithms. MECM can be used with non-metric dissimilarity data. Yet, it still requires to store the whole dissimilarity matrix. Recently, we introduced another evidential clustering procedure, called EK-NNclus [3]. This method uses only the k nearest neighbors of each object: it thus has lower storage requirements than EVCLUS, RECM or MECM, and it is considerably faster. However, it can generate only very simple credal partitions, in which masses are assigned only to singletons $\{\omega_k\}$ and to the set Ω of clusters.

In this paper, we propose two improvements of EVCLUS, which make it applicable to very large dissimilarity datasets. First, the gradient-based optimization procedure in the original EVCLUS algorithm is replaced by an adaptation of the much faster iterative row-wise quadratic programming method proposed in [10]. Secondly, and even more importantly, we show that EVCLUS does not need to be provided with the whole dissimilarity matrix, reducing the time and space complexity from quadratic to roughly linear. The rest of this paper is organized as follows. The basic notions of evidential clustering and the EVCLUS algorithm will first be recalled in Section 2. The improvements to EVCLUS will then be introduced in Section 3, and simulation results will be presented in Section 4. Finally, Section 5 will conclude the paper.

2 Evidential Clustering

The notion of credal partition will first be recalled in Section 2.1. The EVCLUS algorithm will then be summarized in Section 2.2.

2.1 Credal Partition

Assume that we have a set $\mathcal{O} = \{o_1, \ldots, o_n\}$ of n objects, each one belonging to one and only one of c groups or clusters. Let $\Omega = \{\omega_1, \ldots, \omega_c\}$ denote the set of clusters. If we know for sure which cluster each object belongs to, we can provide a partition of the n objects. Such a partition may be represented by binary variables u_{ik} such that $u_{ik} = 1$ if object o_i belongs to cluster ω_k, and $u_{ik} = 0$ otherwise. If objects cannot be assigned to clusters with certainty,

then it is natural to quantify cluster-membership uncertainty by mass functions $m_1, \ldots, m_n$, where each mass function m_i is defined on Ω and describes the uncertainty about the cluster of object i. The n-tuple $\mathcal{M} = (m_1, \ldots, m_n)$ is called a *credal partition* [4]. The notion of credal partition is very general, in the sense that it boils down to several alternative clustering structures when the mass functions composing the credal partition have some special forms [2]. Hard, fuzzy, possibilistic and rough partitions may also be computed from a credal partition as by-products [7, 2]. Recently, evidential clustering has been successfully applied in various domains such as machine prognosis [9], medical image processing [6] and analysis of social networks [11].

2.2 EVCLUS

The EVCLUS algorithm, introduced in [4], constructs a credal partition for dissimilarity data. Let $\boldsymbol{D} = (d_{ij})$ be an $n \times n$ dissimilarity matrix, where d_{ij} denotes the dissimilarity between objects o_i and o_j. Dissimilarities may be distances computed from attribute data, or they may be provided directly, in which case they need not satisfy the axioms of a distance function. To derive a credal partition $\mathcal{M} = (m_1, \ldots, m_n)$ from $\boldsymbol{D}$, we assume that the plausibility pl_{ij} that two objects o_i and o_j belong to the same class is a decreasing function of the dissimilarity d_{ij}: the more similar are two objects, the more plausible it is that they belong to the same cluster. Now, it can be shown [4] that the plausibility pl_{ij} is equal to $1 - \kappa_{ij}$, where κ_{ij} is the *degree of conflict* between m_i and m_j. The credal partition $\mathcal{M}$ should thus be determined in such a way that similar objects have mass functions m_i and m_j with low degree of conflict, whereas highly dissimilar objects are assigned highly conflicting mass functions. This can be achieved by minimizing a *stress function* measuring the discrepancy between the pairwise degrees of conflict and the dissimilarities, up to some increasing transformation. Here, we consider the following stress function,

$$J(\mathcal{M}) = \eta \sum_{i<j} (\kappa_{ij} - \delta_{ij})^2, \tag{1}$$

where $\eta = \left(\sum_{i<j} \delta_{ij}^2\right)^{-1}$ is a normalizing constant, and the $\delta_{ij} = \varphi(d_{ij})$ are transformed dissimilarities, for some fixed increasing function φ from $[0, +\infty)$ to $[0, 1]$. For instance, φ can be chosen as $\varphi(d) = 1 - \exp(-\gamma d^2)$, where γ is a user-defined parameter. Parameter γ can be fixed as follows. For $\alpha \in (0, 1)$, let $d_0 = \varphi^{-1}(1 - \alpha)$ be the dissimilarity value such that two objects whose dissimilarity exceeds d_0 have a plausibility at least equal to $1 - \alpha$. For φ defined as above, we have $\gamma = -\log \alpha / d_0^2$. In the simulations presented in this paper, we used $\alpha = 0.05$, leaving d_0 as the only parameter to be adjusted.

3 Improvements to EVCLUS

In this section, we introduce two improvements to the original EVCLUS algorithm. First, in Section 3.1, we show that the special form of stress function (1)

makes it possible to use an Iterative Row-wise Quadratic Programming (IRQP) algorithm, such as introduced in [10] for latent-class clustering. In Section 3.2, we then propose to use only a subset of the dissimilarities, allowing for a drastic reduction in computing time.

3.1 Optimization algorithm

To simplify the presentation of the IRQP algorithm, let us rewrite (1) using matrix notations. Let us assume that the f focal sets $F_1, \ldots, F_f$ of mass functions $m_1, \ldots, m_n$ have been ordered in some way. We can then represent each mass function m_i by a vector $\boldsymbol{m}_i = (m_1(F_1), \ldots, m_i(F_f))^T$ of length f. The credal partition $\mathcal{M} = (m_1, \ldots, m_n)$ can then be represented by a matrix $\boldsymbol{M} = (\boldsymbol{m}_1^T, \ldots, \boldsymbol{m}_n^T)^T$ of size $n \times f$. The degree of conflict between two mass functions m_i and m_j can be written as $\kappa_{ij} = \boldsymbol{m}_i^T \boldsymbol{C} \boldsymbol{m}_j$, where $\boldsymbol{C}$ is the square matrix of size f, with general term $C_{k\ell} = 1$ if $F_k \cap F_\ell = \emptyset$ and $C_{k\ell} = 1$ otherwise. With these notations, the stress function (1) can be written as

$$J(\boldsymbol{M}) = \eta \sum_{i<j} (\boldsymbol{m}_i^T \boldsymbol{C} \boldsymbol{m}_j - \delta_{ij})^2. \tag{2}$$

In [4], we proposed to minimize J using a gradient-based algorithm. Another approach, which better exploits the particular form of (1), is to minimize $J(\boldsymbol{M})$ with respect to each row of $\boldsymbol{M}$ at a time, keeping the other rows constant [10]. Minimizing $J(\boldsymbol{M})$ with respect to $\boldsymbol{m}_i$ is equivalent to minimizing

$$g(\boldsymbol{m}_i) = \|\boldsymbol{M}_{-i} \boldsymbol{C} \boldsymbol{m}_i - \boldsymbol{\delta}_i\|^2, \tag{3}$$

where $\boldsymbol{M}_{-i}$ is the matrix obtained from $\boldsymbol{M}$ by deleting row i, and $\boldsymbol{\delta}_i$ is the vector of transformed dissimilarities δ_{ij} between object o_i and all other objects o_j, $j \neq i$. Minimizing $g(\boldsymbol{m}_i)$ under the contraints $\boldsymbol{m}_i^T \boldsymbol{1} = 1$ and $\boldsymbol{m}_i \geq \boldsymbol{0}$ is a linearly constrained positive least-squares problem, which can be solved using efficient algorithms. By iteratively updating each row of $\boldsymbol{M}$ as described above, as long as the overall function value decreases, the algorithm converges to a stable function value, which is at least a local minimum.

3.2 *k*EVCLUS

As mentioned in Section 1, the $O(n^2)$ complexity of EVCLUS, where n is the number of objects, makes it inapplicable to large dissimilarity data. The fundamental reason for this high complexity is the fact that the calculation of stress criterion (1) requires the full dissimilarity matrix. However, there is usually some redundancy in a dissimilarity matrix, even if the dissimilarity measure is not a distance. In particular, if two objects o_1 and o_2 are very similar, then any object o_3 that is dissimilar from o_1 is usually also dissimilar from o_2. Because of such redundancies, it might be possible to compute the differences between degrees of conflict and dissimilarities, for *only a subset of randomly sampled dissimilarities.*

More precisely, let $j_1(i), \ldots, j_k(i)$ be k integers sampled at random from the set $\{1, \ldots, i-1, i+1, \ldots, n\}$, for $i = 1, \ldots, n$. Let J_k the following stress criterion,

$$J_k(\mathcal{M}) = \eta \sum_{i=1}^{n} \sum_{r=1}^{k} (\kappa_{i,j_r(i)} - \delta_{i,j_r(i)})^2, \tag{4}$$

where, as before, η is a normalizing constant, $\eta = \left(\sum_{i,r} \delta^2_{i,j_r(i)}\right)^{-1}$. Obviously, $J(\mathcal{M})$ is recovered as a special case when $k = n - 1$. However, in the general case, the calculation of $J_k(\mathcal{M})$ requires only $O(nk)$ operations. If k can be kept constant as n increases, or, at least, if k increases slower than linearly with n, then significant gains in computing time and storage requirement could be achieved. In the experiments below, we show that this version of EVCLUS (hereafter referred to as k-EVCLUS) is more scalable than the original version, and applicable to large dissimilarity datasets.

4 Experiments

In this section, we first report some results showing the superiority of IRQP over the gradient-based optimization procedure in Section 4.1. Experiments with k-EVCLUS are then reported in Section 4.2. For all the experiments reported in this section, we used the version of EVCLUS with the empty set $\emptyset$, the singletons $\{\omega_k\}$, and Ω as focal sets. The k-EVCLUS algorithm, as well as other evidential clustering procedures, has been implemented in the R package[4] `evclust` [1].

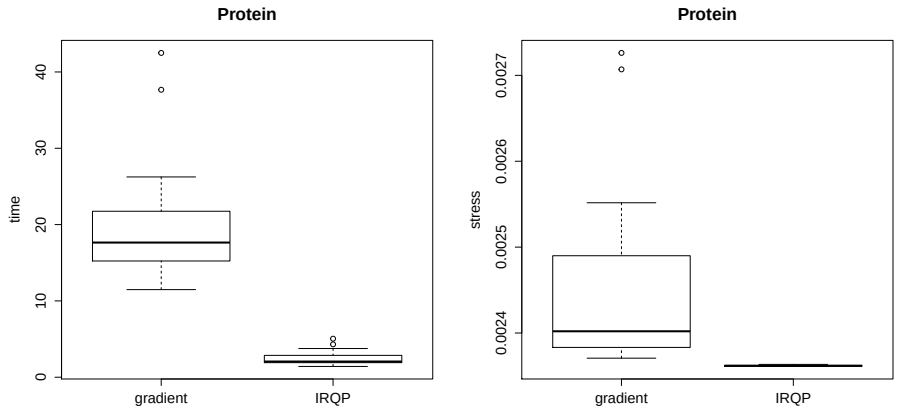
4.1 Comparison between IRQP and gradient-based optimization

The Protein dataset [4] consists of a dissimilarity matrix derived from the structural comparison of 213 protein sequences. Each of these proteins is known to belong to one of four classes of globins. We ran the Gradient and IRQP algorithms on the Protein dataset with $c = 4$, and parameter d_0 set to the largest dissimilarity value. Both algorithms were run 20 times from 20 random initial values. In each run, both algorithms were started from the same initial conditions. Figure 1, which shows boxplots of the stress values at convergence and computing times, for both algorithms. We can see that, on this data, the IRQP algorithm converges more than 10 times faster than the Gradient algorithm. The stress values at convergence for IRQP also have lower variability and are consistently smaller than those obtained by the Gradient algorithm.

4.2 Evaluation of k-EVCLUS

In this section, we report experiments with artificial datasets composed of four clusters of $n/4$ two-dimensional vectors, generated from a multivariate t distribution with five degrees of freedom and centered, respectively, on $[0, 0]$, $[0, 5]$, $[5, 0]$

[4] Available from the CRAN web site at `https://cran.r-project.org/web/packages`.



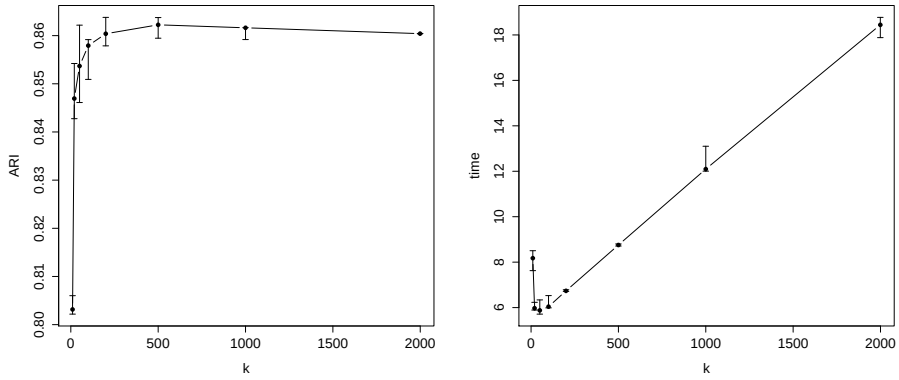
(a) (b)

Fig. 1. Boxplots of computing time (a) and stress value at convergence (b) for 20 runs of the Gradient and IRQP algorithms on the Protein data.

and $[5, 5]$. The dissimilarities were computed as the Euclidean distances between the data points. Algorithm k-EVCLUS was run with d_0 equal to the 0.9-quantile of distances and $c = 4$. Figure 2 shows the Adjusted Rand Index (ARI) and computing time[5] as functions of k for a simulated dataset with $n = 2000$. The ARI was computed after transforming the credal partition into a hard partition by selecting, for each object, the cluster with the largest plausibility. The values of k were chosen as 10, 20, 50, 100, 200, 500 and 1999. When $k = 1999 = n-1$, the whole distance matrix is used, and k-EVCLUS boils down to EVCLUS. As we can see, k-EVCLUS performs as well as EVCLUS ($k = 1999$) according to the ARI (Figures 2(a)), as long as $k \geq 100$, with a significant gain in training time (Figure 2(b)). We observe that the computing time is higher for $k = 10$ than it is for $k = 20$, which is due to the fact that the algorithm took more time to converge for $k = 10$.

To compare k-EVCLUS with RECM and EK-NN on this clustering problem, we let n vary in from 1000 to 5000 (by 1000 increments), and we generated 10 datasets of each size, from the same distribution. We then recorded the computing times and ARI values for for k-EVCLUS (with $k = 100$ and d_0 equal to the 0.9-quantile of the distances), RECM (with the same parameters as above), and EK-NNclus with $K = 3\sqrt{n}$ and $q = 0.95$. The results are reported in Figure 3. From Figure 3(a), we can see that k-EVCLUS and EK-NNclus are comparable in terms of computing time for different values of n, whereas the time complexity of RECM seems to be considerably higher. On the other hand, k-EVCLUS and RECM yield comparable results in terms of ARI (see Figure 3(b)), whereas the

[5] All simulations reported in this paper were performed on an Apple MacBook Pro computer with a 2.5 GHz Intel Core i7 processor.



(a) (b)

Fig. 2. Adjusted Rand Index (a) and computing time (b) of k-EVCLUS as a function of k, as a function of k, for the simulated data with $n = 2000$. The error bars show the median as well as the lower and upper quartiles over 10 runs of the algorithm.

partitions obtained by EK-NNclus have higher variability. It must be noticed that the number c of clusters is specified for k-EVCLUS and RECM, but it is not for EK-NNclus. Overall, k-EVCLUS seems to provide the best results (for correctly specified c) in the least amount of time. More extensive results with several synthetic and real datasets are reported in [5].

5 Conclusions

In its original version, EVCLUS was significantly slower than more recently introduced relational evidential clustering algorithms such as RECM and EK-NNclus. Also, it was limited to datasets of a few thousand objects, due to the necessity to store the whole dissimilarity matrix. In this paper, we have been able to overcome these limitations, thanks to two major improvements. First, the original gradient algorithm has been replaced by a much more efficient iterative row-wise quadratic programming procedure, which exploits the particular structure of the optimization problem. Secondly, we have shown that we only need to supply EVCLUS with the dissimilarities between each object and k randomly selected objects, reducing the space complexity from $O(n^2)$ to $O(kn)$. The improvements described in this paper make EVCLUS potentially applicable to large dissimilarity data, with of the order of 10^4 or even 10^5 objects. Analyzing even larger datasets (with millions of objects, as arising in social network studies, for instance), would probably require to sample the rows of the dissimilarity matrix. This issue requires further investigation.

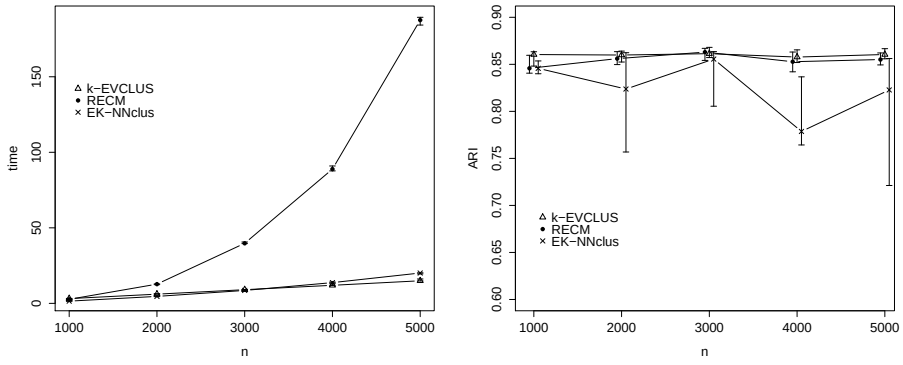


Fig. 3. Computing time (a) and ARI (b) for k-EVCLUS, RECM and EKNNclus for simulated datasets with different values of n.

References

1. T. Denœux. *evclust: Evidential Clustering*, 2016. R package version 1.0.2.
2. T. Denoeux and O. Kanjanatarakul. Beyond fuzzy, possibilistic and rough: An investigation of belief functions in clustering. In *8th International Conference on Soft Methods in Probability and Statistics (SMPS 2016)*, Rome, Italy, Sept. 2016.
3. T. Denœux, O. Kanjanatarakul, and S. Sriboonchitta. EK-NNclus: a clustering procedure based on the evidential k-nearest neighbor rule. *Knowledge-based Systems*, 88:57–69, 2015.
4. T. Denœux and M.-H. Masson. EVCLUS: Evidential clustering of proximity data. *IEEE Trans. on Systems, Man and Cybernetics B*, 34(1):95–109, 2004.
5. T. Denœux, S. Sriboonchitta, and O. Kanjanatarakul. Evidential clustering of large dissimilarity data. *Knowledge-based Systems*, 106:179–195, 2016.
6. B. Lelandais, S. Ruan, T. Denœux, P. Vera, and I. Gardin. Fusion of multi-tracer PET images for dose painting. *Medical Image Analysis*, 18(7):1247–1259, 2014.
7. M.-H. Masson and T. Denoeux. ECM: an evidential version of the fuzzy c-means algorithm. *Pattern Recognition*, 41(4):1384–1397, 2008.
8. M.-H. Masson and T. Denœux. RECM: relational evidential c-means algorithm. *Pattern Recognition Letters*, 30:1015–1026, 2009.
9. L. Serir, E. Ramasso, and N. Zerhouni. Evidential evolving Gustafson-Kessel algorithm for online data streams partitioning using belief function theory. *International Journal of Approximate Reasoning*, 53(5):747–768, 2012.
10. C. J. ter Braak, Y. Kourmpetis, H. A. Kiers, and M. C. Bink. Approximating a similarity matrix by a latent class model: A reappraisal of additive fuzzy clustering. *Computational Statistics & Data Analysis*, 53(8):3183–3193, 2009.
11. K. Zhou, A. Martin, Q. Pan, and Z.-G. Liu. Median evidential c-means algorithm and its application to community detection. *Knowledge-Based Systems*, 74(0):69–88, 2015.