Weighted Similarity Majority Margins Based Multiple Attributes Clustering

Raymond Bisdorff[1], Patrick Meyer[23], and Alexandru-Liviu Olteanu[123]

[1] CSC/ILIAS, FSTC, University of Luxembourg
[2] Institut Télécom, Télécom Bretagne, UMR CNRS 3192 Lab-STICC, Technopôle
Brest Iroise, CS 83818, 29238 Brest Cedex 3, France
[3] Université Européenne de Bretagne

Abstract. We propose a meta-heuristic for clustering objects that are described on multiple incommensurable attributes of nominal, ordinal and/or cardinal type. Our approach makes use of an innovative bipolar-valued dual similarity-dissimilarity relation characterized by pairwise weighted majority margins of similar minus dissimilar attribute evaluations. The clustering is computed in two steps. First, an evolutionary algorithm searches for a suitable subset of maximal similarity cliques that will best serve as cluster cores. In a second step, we construct with a greedy heuristic, around these initial cluster cores, a corresponding final partition which best fits the given bipolar-valued similarity relation.

keywords: multiple incommensurable attributes clustering, weighted majority margins, bipolar-valued dual similarity-dissimilarity relation, maximal similarity cliques, evolutionary algorithm

1 Introduction

Clustering can be defined as an unsupervised grouping method that gathers objects that are similar and separates those that are not. Unlike classification, clustering has no a priori information regarding the groups to which to assign the objects to be clustered. It is actually widely used in many fields like artificial intelligence, information technology, image processing, biology, psychology, marketing and others. Many clustering algorithms have been developed to suit the various requirements that each application field has regarding the expected clustering results. Clustering methods can be categorized into partitioning [2, 23, 22, 29], hierarchical [13, 14, 16, 19, 31], density-based [3, 10, 11, 32], grid-based [1, 30] and model-based methods [8, 18, 21, 28]. Jain gives a thorough presentation of many of these methods in [17]. The emerging field of community detection has recently brought forth new graph-based methods of clustering [9, 25–27]. Fortunato covers thoroughly many of the latest ones developed in this field [12].

Here we propose a method for partitioning objects that are described by multiple weighted and incommensurable attributes of nominal, ordinal and or cardinal type. Our work is inspired by the bipolar outranking approach proposed

by [4–6] for dealing with multiple criteria decision aid problems. We first characterize pairwise global similarity statements by balancing marginal similarity and dissimilarity situations observed at attribute level in order to get majority margins, i.e. a bipolar-valued similarity graph. Interesting maximal cliques of objects in this graph, chosen especially for their particular fitness as potential cluster cores, are selected and then expanded to form a complete partition. As the enumeration of all the maximal cliques is well known to be potentially exponential [24], we develop a special meta-heuristic for dealing with the selection of the most promising ones.

2 Dual similarity-dissimilarity modelling

To illustrate the relational concepts of similarity and dissimilarity we first present a small didactic problem.

Example 1. Let us consider in Table 1 a set of objects $\{a, b, c, d\}$ that are described in each column by four values observed on attributes $\{1, 2, 3, 4\}$ with respect to a common ratio scale from 0 to 100. We may notice that objects a, b and c have similar values

Table 1. Objects' evaluations on the four attributes

Attributes	a	b	c	d
1	10	15	20	93
2	32	38	50	55
3	44	40	30	20
4	66	82	58	50

on the first attribute, while they are all very different from object d. On the second attribute a and b, as well as c and d have close evaluations. On the third attribute a and d have slightly different values while b and c show quite similar evaluations. And on the last attribute, only object b is considerably different from the rest. Two objects are considered to be similar if they have similar evaluations on a majority of attributes. For example objects a and b have close evaluations on three out of four attributes, therefore they are considered to be globally *similar*. Objects c and d have also three attributes out of four on which they are similar. But on the first attribute, they show a very large difference in evaluations (93 compared to 20). Here, we would rather like to say that *we don't know* if they are similar or not.

2.1 Pairwise similarity and dissimilarity statements

Let $X = \{x, y, z, ...\}$ denote a set of n objects. Each object $x \in X$ is described on a set $I = \{i, j, k, ...\}$ of m attributes of nominal, ordinal and/or cardinal type. For any $x \in X$ and $i \in I$, the actual evaluation x_i may be encoded without loss of generality in a real range $[m_i, M_i]$. The attributes may not all be of the same

significance for assessing the global similarity between the objects. Therefore we assign to the attributes normalized weights $w_i \in [0,1]$ s.t. $\sum_{i \in I} w_i = 1$.

In order to characterize a *marginal pairwise similarity* relation between two alternatives x and y of X for each attribute i of I , we use the function $s_i :$ $X \times X \to \{-1, 0, 1\}$ defined as follows:

$$s_i(x,y) := \begin{cases} -1 & \text{, if } |x_i - y_i| \geqslant \delta_i; \\ +1 & \text{, if } |x_i - y_i| \leqslant \sigma_i; \\ 0 & \text{, otherwise.} \end{cases} \tag{1}$$

where $0 \leq \sigma_i < \delta_i \leq M_i - m_i, \forall i \in I$ denote marginal similarity and dissimilarity discrimination thresholds. These thresholds may be constant and/or proportional to the values taken by the objects being compared. If $s_i(x,y) = +1$ (respectively $s_i(x,y) = -1$) we conclude that x and y are similar (respectively not similar) on attribute i. When $s_i(x,y) = 0$ we are in doubt whether x and y are, on attribute i, to be considered similar or not similar.

Example 2. In Example 1, if we set the similarity (resp. dissimilarity) thresholds on attribute 1 to $\sigma_1 = 10$ (resp. $\delta_1 = 20$), we notice in Table 1 that the absolute difference between the evaluations of objects a and b is quite small ($|a_1 - b_1| = 5 < \sigma_1$). Therefore $s_1(a,b) = +1$, and we consider that the objects a and b are similar on attribute 1.

The *weighted similarity* relation between x and y, aggregating all marginal similarity statements is characterized via the function $ws : X \times X \to [-1, 1]$ defined as follows:

$$ws(x,y) := \sum_{i \in I} w_i \cdot s_i(x,y) \tag{2}$$

Again, if $0 < ws(x,y) \leqslant 1$ we may assume that it is more sure than not that x is similar to y; if $-1 \leqslant ws(x,y) < 0$ we may assume that it is more sure that x is not similar to y than the opposite; if, however, $ws(x,y) = 0$ we are in doubt whether object x is similar to object y or not.

Example 3. If we consider that in Example 1, all four attributes are equally significant, i.e. $w_i = 0.25, i = 1,..4$, we may notice in Table 1 that $s_1(a,b) = 1$, $s_2(a,b) = 1$, $s_3(a,b) = 1$ and $s_4(a,b) = 0$. Therefore, $ws(a,b) = 0.75$, and we may conclude that objects a and b, when described by these attributes, are more similar than not.

In the same manner we can characterize the *marginal dissimilarity* of two objects on attribute i via the function $d_i : X \times X \to \{-1, 0, 1\}$ defined as follows for each i of I:

$$d_i(x,y) := \begin{cases} -1 & \text{, if } |x_i - y_i| \leqslant \sigma_i; \\ +1 & \text{, if } |x_i - y_i| \geqslant \delta_i; \\ 0 & \text{, otherwise.} \end{cases} \tag{3}$$

If $d_i(x,y) = +1$ (resp. $d_i(x,y) = -1$) we say that x and y are dissimilar (resp. not dissimilar) on attribute i. When $d_i(x,y) = 0$ we are in doubt whether x and y are, on attribute i, to be considered dissimilar or not dissimilar. The *weighted*

dissimilarity relation between x and y, aggregating all marginal dissimilarity situations, is characterized via the function $wd : X \times X \to [-1, 1]$ as follows:

$$wd(x, y) := \sum_{i \in I} w_i \cdot d_i(x, y) \tag{4}$$

Property 1. The weighted dissimilarity is the *negation* of the weighted similarity relation: $wd = -ws$.

Property 1 follows immediately from Formulas (1) and (3) where we can easily verify that $s_i(x, y) = -d_i(x, y)$, $\forall i \in I$, $\forall x, y \in X$.

2.2 Taking into account strong dissimilarities

As we have seen in Example 1, in some cases two objects may be similar on most of the attributes but, show, nonetheless, a very strong dissimilarity on some other attribute. In this case the objects can hardly be considered to be on the overall similar or dissimilar. To model this *indeterminate* situation, we define a *marginal strong dissimilarity* relation between objects x and y with the help of function $sd_i : X \times X \to \{0, 1\}$ as follows:

$$sd_i(x, y) := \begin{cases} 1 & \text{, if } |x_i - y_i| \geq \delta_i^+; \\ 0 & \text{, otherwise.} \end{cases} \tag{5}$$

where δ_i^+ is such that $\delta_i < \delta_i^+ \leqslant M_i - m_i$ and represents a strong dissimilarity discriminating threshold. If $sd_i(x, y) = 1$ (resp. $sd_i(x, y) = 0$) we conclude that x and y are *strongly dissimilar* (resp. not strongly dissimilar) on attribute i.

Example 4. If we set in Example 1 the strong dissimilarity threshold δ_1^+ to 70, we may notice in Table 1 that objects c and d present such a large difference on the first attribute, $|c_1 - d_1| = 73 > \delta_1^+$. Therefore, $sd_1(c, d) = 1$ and we conclude that they are strongly dissimilar on this attribute.

Definition 1 (Overall similarity relation). *We consider that two objects x and y of X, described on a set I of attributes, are* similar*, denoted $(x \, \mathsf{S} \, y)$, if*

1. *a weighted majority of the attributes in I validates a similarity situation between x and y and,*
2. *there is no marginal strong dissimilarity situation observed between x and y.*

We formally characterize such an *overall similarity* relation by function $s : X \times X \to [-1, 1]$ as follows:

$$s(x, y) \quad := \quad \bigvee \left(ws(x, y), -sd_1(x, y), \cdots, -sd_m(x, y) \right) \tag{6}$$

where, for $q \in \mathbb{N}_0$, the epistemic disjunction operator $\bigvee : [-1, 1]^q \to [-1, 1]$ is defined as follows:

$$\bigvee (p_1, p_2, \ldots, p_q) := \begin{cases} \max(p_1, p_2, \ldots, p_q) & \text{, if } p_i \geqslant 0, \ \forall i \in \{1 \ldots q\}; \\ \min(p_1, p_2 \ldots, p_q) & \text{, if } p_i \leqslant 0, \ \forall i \in \{1 \ldots q\}; \\ 0 & \text{, otherwise.} \end{cases} \tag{7}$$

Indeed, for two given alternatives x and y of X, if $ws(x, y) > 0$ and no marginal strong dissimilarity has been detected, $ws(x, y) = s(x, y)$ and both alternatives are considered as similar. If however $ws(x, y) > 0$ and a strong dissimilarity is detected, to be prudent, we do not state that x and y are similar or dissimilar, and $s(x, y) = 0$. Now, if $ws(x, y) < 0$ and, additionally, a marginal strong dissimilarity is observed, then the dissimilarity between x and y is certainly confirmed and $s(x, y) = -1$. Finally, if $ws(x, y) = 0$ is observed conjointly with a strong dissimilarity, we will conclude that x and y are indeed dissimilar and $s(x, y)$ is put to -1.

Example 5. If we look again at objects c and d in Example 1, we notice that they can be considered as similar according to the weighted similarity $(ws(c, d) = 0.75)$. However, they show a strong dissimilarity on the first attribute, $(sd_1(c, d) = 1)$. Therefore, we cannot decide whether they are similar or not and $s(x, y) = 0$.

Likewise we can characterize an *overall dissimilarity* relation in accordance with Definition 1 via function $d : X \times X \to [-1, 1]$ defined as follows:

$$d(x, y) \quad := \quad \bigvee \left(wd(x, y), sd_1(x, y), \cdots, sd_m(x, y) \right) \tag{8}$$

Property 2 (Overall similarity-dissimilarity duality).
The overall dissimilarity represents the *negation* of the overall similarity:
$d = -s$.

Property 2 results from Property 1 and Definition (7) of the epistemic disjunction operator.
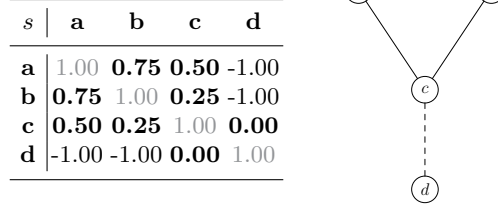
2.3 The Condorcet similarity graph

We call *Condorcet similarity graph*, denoted $G(X, s^*)$, the three-valued graph associated with the bipolar-valued similarity relation s, where X denotes the set of nodes and function $s^* : X \times X \to \{-1, 0, 1\}$ characterizes its set of edges as follows:

$$s^*(x, y) \quad := \quad \begin{cases} +1 & \text{, if } s(x, y) > 0; \\ -1 & \text{, if } s(x, y) < 0; \\ 0 & \text{, otherwise.} \end{cases} \tag{9}$$

Furthermore, following Definition 1, we may denote S the set $\{(x, y) \in X \times X \text{ s.t. } s(x, y) > 0\}$ of pairs of *overall similar* objects.

Example 6. Figure 1 represents on the left the bipolar-valued overall similarity relation s related to Example 1. Notice that a,b and c are more similar than not to each other, whereas d is surely dissimilar both from a and b. Besides, d and c appear to be neither similar nor dissimilar. The corresponding Condorcet similarity graph is shown on the right in Figure 1. Edges valued by -1 are not represented and the zero-valued one is dashed. As a Condorcet similarity graph is always reflexive, we do not represent the loops on the nodes.

Fig. 1. Bipolar-valued similarity relation (left) with associated Condorcet similarity graph (right)

s	**a**	**b**	**c**	**d**
a	1.00	**0.75**	**0.50**	-1.00
b	**0.75**	1.00	**0.25**	-1.00
c	**0.50**	**0.25**	1.00	**0.00**
d	-1.00	-1.00	**0.00**	1.00

3 Definition of the clusters

Ideally, a cluster of objects would have all the objects inside it similar to each other and dissimilar from the rest. In graph theory this may be modeled by the maximal clique construction, which is a fully connected set of objects to which no other object can be added without breaking this property. However, we would also need the maximal clique to be totally disconnected from the rest of the graph, which on real clustering data will very rarely be the case. Generally, maximal cliques are more or less connected to other objects. Also there may generally exist a very large number of such maximal cliques, many overlapping one with the other. Moon and Moser have shown that, in the worst case, the number of maximal cliques in a graph can be exponential [24]. Nonetheless, such well disconnected maximal cliques, if they exist, are first class candidates for the clusters we are looking for.

Therefore, in a first stage, we propose to select in the overall similarity graph $G(X, S)$ the *best* set of maximal cliques that may be considered as cluster cores. In a second stage, we expand then these cluster cores to eventual clusters – the actual partition parts – by adding objects that are well connected to them. This set of maximal cliques can contain both well disconnected maximal cliques which can be on their own considered as clusters as well as maximal cliques that are well connected to pretty much the same set of objects and can be considered as cluster cores.

Let us first introduce a fitness measure for estimating the quality of a maximal clique to serve as cluster core. Given a Condorcet similarity graph $G(X, s^*)$ and a set $C \subseteq X$ of objects, we define, for each x of X the strength of its connectivity to C as:

$$c_C(x) \ := \ \frac{\sum_{y \in C} s^*(x, y)}{|C|}. \tag{10}$$

The profile of C is then defined by the set of all possible connectivity strength values. If the set of objects C shows a uniform trend in the way it compares to the other objects in the data-set, then the profile of C would contain values close to either $+1$ or -1. We say that such a profile is strong and define as follows the

fitness $f(C)$ of C to serve as cluster core :

$$f(C) \; := \; \frac{\sum_{x \in X} |c_C(x)|}{|X|} \tag{11}$$

Example 7. In Figures 2 and 3, we show how this measure can characterize possible cluster cores. All these Figures show a Condorcet similarity graph defined on a same set of 10 objects. Figure 2 shows a good cluster core where the maximal clique $\{1, 2, 3, 4, 5\}$ is well connected to objects 6 and 7, and very well disconnected from the rest of the objects. Objects 6 and 7 are connected to 4 out of 5 objects in the maximal clique and can be added later to eventually form a cluster. Whereas Figure 3 shows a maximal
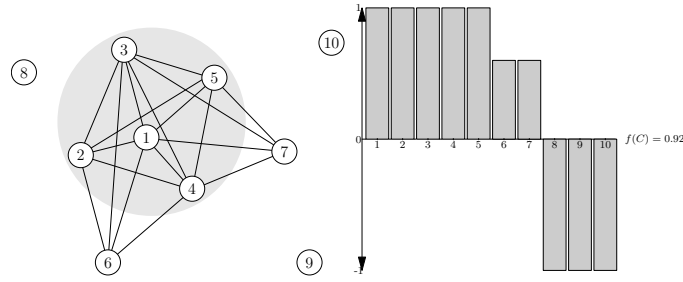

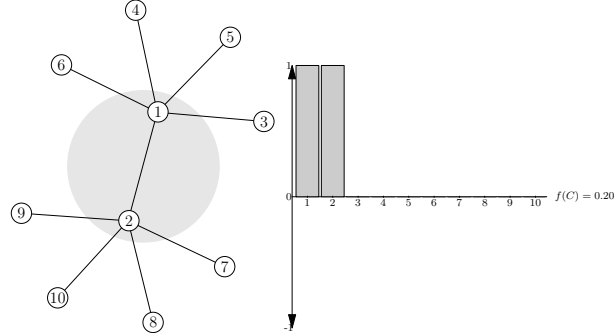**Fig. 2.** Good Cluster Core Example

Fig. 3. Bad Cluster Core Example

clique that gives a bad cluster core and this is clearly reflected in its fitness measure.

The best maximal cliques with respect to our fitness measure will most of the times be concentrated in a same area of the Condorcet similarity graph. In order

to achieve a partitioning of the entire data-set we need to detect well separated maximal cliques that correspond in fact to local maxima of our fitness measure. To find these local maxima, we define the neighborhood of a set as all the sets that contain at least one object from the first:

$$N(C) \; = \; \{K \subset X \mid \forall x, y \in K : \; s(x,y) > 0, \; \exists z \in K : z \in C\} \qquad (12)$$

C is a suitable cluster core if it is a maximal clique and its fitness is better than that of the maximal cliques inside its neighborhood, i.e. $f(K) \leqslant f(C), \forall K \in N(C)$.

Now, after finding in a first step, suitable cluster cores with the help of the fitness measure shown in Equation 11, we partition the set X by adding the objects that were left out. For this we need to define the similarity of an object $x \in X$ to a set $C \subseteq X$ with the help of the following function:

$$s_C(x) := \frac{\sum_{y \in C} s(x,y)}{|C|} \quad . \qquad (13)$$

4   Clustering Algorithm

We structure our algorithmic approach in three steps:

1. First, we construct the bipolar-valued similarity relation and its associated Condorcet similarity graph. This step is in $\mathcal{O}(n^2 m)$, deriving directly from the definitions we have given before.
2. Then, we compute the cluster cores from the fittest maximal cliques of the Condorcet similarity graph.
3. Finally, we assign in a greedy heuristic way all the objects that were left out to the cluster core to which they are most similar to.

In the second step, we may use two resolution strategies: exact enumeration of all the maximal cliques and selection of the fittest ones as potential cluster cores, or, a population based meta-heuristic approach.

For the exact approach we use the Bron-Kerbosch algorithm [7], with the pivot point improvement from Koch [20] to reduce the number of recursive calls of the procedure. We then evaluate the fitness of each maximal clique and compute the neighbourhood matrix from which we retrieve the maximal cliques that are the local maxima of the fitness function. As previously mentioned, the number of maximal cliques in a graph can be exponential [24], making the use of exact approaches for large or even medium clustering problems rapidly intractable.

To overcome this operational problem, we use a population-based meta-heuristic close in structure to evolutionary strategies. Each individual in the population is a maximal clique in the Condorcet similarity graph. Our aim is to eventually discover all maximal cliques that are local maxima of our fitness measure (11).

In the *initialization step* we, first, iteratively generate maximal cliques that do not overlap with each other. After each object has been covered by at least

one maximal clique, the rest of the population is then generated randomly. We acknowledge the fact that this step could be improved further by opting for objects with many similarity links to build the first maximal cliques and thus cover the data-set quicker. The number of maximal cliques that cover the data-set could also be an indicator of the needed size of the population.

The *selection step* has a large number of potential variations. As examples, we may mention the uniform selection, roulette wheel, stochastic universal sampling and tournament selection. They can use either the fitness measure or, in case the fitness values are very close, the fitness rank. We have opted after several tests for the rank-based roulette wheel method.

The *reproduction step* is based on a mutation operator specifically designed for maximal cliques. The maximal clique that will generate a new individual in the population is incrementally stripped with a given probability of its objects and then grown by adding other objects until the property of maximality is reached. The generated population is of equal size with the old one.

In the *replacement step*, all maximal cliques in the current population that are local maxima of the fitness measure, based however on the limited exploration of their neighborhoods that has been done at previous iterations, are kept in the new population. The rest of the individuals to be kept are selected at random. The information gathered at each iteration regarding the neighborhood of each maximal clique is stored as the best fitness value each object had at some point as part of a maximal clique.
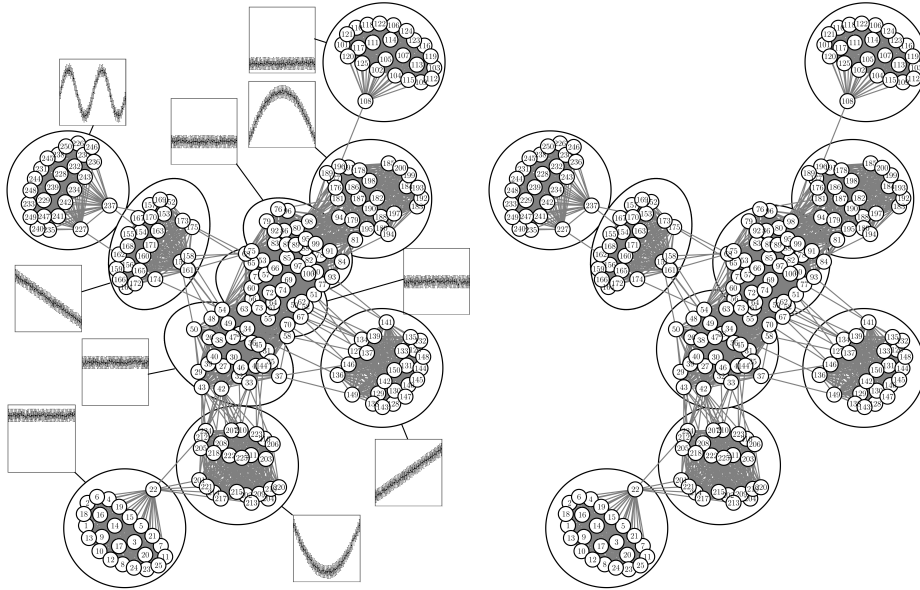

Fig. 4. Clustering Benchmark: Original classes (left); Clustering result (right)

In order to test our clustering algorithm and compare it with other clustering approaches, we have generated a benchmark set of 180 problem instances. These include instances with 100, 250, 500 and 1000 objects, defined on 10, 30 or 50 attributes. The attributes are defined on commensurable ratio scales from 0 to 100 points. The objects are generated from 10 profile objects by randomly shifting their attribute values by as much as ± 5, ± 10 or ± 15. Similarity and dissimilarity discrimination thresholds are set to 10, respectively 20 points.

In Figure 4 we show on the left the Condorcet similarity graph of one benchmark instance with 250 objects outlining the 10 equally sized classes of objects and boxplots of their evaluations. On the right we show the result of our clustering algorithm. All 10 classes are faithfully clustered, except the class in the middle where the profiles are very close to the neighbor classes and may contain more or less similar objects.

In Table 2, we present the average results of our algorithm compared with the results from the classical K-means [22] and Agglomerative Hierarchical Clustering (AHC) [16] algorithms. For the latter we used the single link variant. Both of these algorithms were given the a priori knowledge that the results should contain 10 clusters. These results come from running every algorithm 10 times on each of the 180 benchmarks. Our algorithm was running on the 100, 250, 500 and 1000 object instances for 1, 2, 3, and 5 minutes respectively. For all of them we used a population size of 20.

Table 2. Algorithmic results

Clustering algorithms	Jaccard Coeff.	Graph Distance
K-means	0.650	8.0%
AHC SL	0.946	4.0%
Our algorithm	0.784	3.8%

The *Jaccard Coefficient* [15] in the second column of Table 2 measures how close the results of the clustering algorithms are to the original classes. It is valued on a scale from 0 to 1, where a value of 1 would mean the resulting clusters are exactly the original classes. Our algorithm does much better than K-means, but less than AHC. However, we have to consider that we neither need to provide commensurable cardinal attributes nor an a priori number of clusters.

The *Graph Distance* shown in the last column, measures the difference between the original Condorcet similarity graph and the graph implied by the clustering result. This latter graph is constructed by placing arcs between all objects inside the same cluster and placing non-arcs between objects inside different clusters. This measure shows the percentage of differences between the two graphs. Here our algorithm is again better than K-means, but also slightly better than AHC.

5 Conclusions and Perspectives

We conclude from the testing above that our clustering method does indeed give consistent classification results, comparable with algorithms like K-means and AHC-SL, however without any requirements on the data, as all kinds of attribute types can be considered. Furthermore, imprecision, uncertainties and even missing values can easily be handled by the similarity relation defined in this article. There are many improvements that could be done to increase the performance of our approach, which will be explored in the future. At the moment there is still a need to experiment with different variations in the meta-heuristic to assure a faster convergence. The final result could be further improved by means of a local search method. Also, the way in which the objects outside the cluster cores are assigned to the clusters should be analyzed more precisely. We also plan to explore the influence of variations of the discrimination thresholds (and the inclusion of proportional thresholds) on the outcomes.

References

1. R. Agrawal, J. Gehrke, D. Gunopulos, and P. Raghavan. Automatic subspace clustering of high dimensional data, 2005.
2. M. R. Anderberg. *Cluster Analysis for Applications*. Academic Press, 1973.
3. M. Ankerst, M. M. Breunig, H. Kriegel, and J. Sander. Optics: ordering points to identify the clustering structure. In *International Conference on Management of Data*, pages 49–60, 1999.
4. R. Bisdorff. Logical foundation of fuzzy preferential systems with application to the electre decision aid methods. *Computers & Operations Research*, 27:673–687, 2000.
5. R. Bisdorff. Electre-like clustering from a pairwise fuzzy proximity index. *European Journal of Operational Research*, 138(2):320–331, 2002.
6. R. Bisdorff. On clustering the criteria in an outranking based decision aid approach. In *Modelling, Computation and Optimization in Information Systems and Management Sciences*, pages 409–418. Springer CCIS, 2008.
7. C. Bron and J. Kerbosch. Algorithm 457: finding all cliques of an undirected graph. *Commun. ACM*, 16(9):575–577, 1973.
8. P. Cheeseman and J. Stutz. *Bayesian Classification (AutoClass): Theory and Results*, chapter 6, pages 62–83. AAAI Press/MIT Press, 1996.
9. Nan Du, Bin Wu, Xin Pei, Bai Wang, and Liutong Xu. Community detection in large-scale social networks. In *WebKDD/SNA-KDD '07: Proceedings of the 9th WebKDD and 1st SNA-KDD 2007 workshop on Web mining and social network analysis*, pages 16–25. ACM, 2007.
10. L. Ertoz, M. Steinbach, and V. Kumar. Finding clusters of different sizes, shapes, and densities in noisy, high dimensional data. In Daniel Barbara and Chandrika Kamath, editors, *Proceedings of the Third SIAM International Conference on Data Mining (SDM 2003)*, volume 112 of *Proceedings in Applied Mathematics*. Society for Industrial and Applied Mathematics, 2003.
11. M. Ester, H. Kriegel, J. Samder, and X. Xu. A density-based algorithm for discovering clusters in large spatial databases with noise. In *Proc. of 2nd International Conference on Knowledge Discovery and*, pages 226–231, 1996.

12. S. Fortunato. Community detection in graphs. *Physics Reports*, 486(3-5):75 – 174, 2010.

13. S. Guha, R. Rastogi, and K. Shim. CURE: an efficient clustering algorithm for large databases. In Laura Haas, Pamela Drew, Ashutosh Tiwary, and Michael Franklin, editors, *SIGMOD '98: Proceedings of the 1998 ACM SIGMOD International Conference on Management of Data*, pages 73–84. ACM Press, 1998.

14. S. Guha, R. Rastogi, and K. Shim. Rock: A robust clustering algorithm for categorical attributes. In *ICDE '99: Proceedings of the 15th International Conference on Data Engineering*, page 512. IEEE Computer Society, 1999.

15. P. Jaccard. Nouvelles recherches sur la distribution florale. *Bulletin de la Societe Vaudoise de Sciences Naturelles*, 44:223–370, 1908.

16. A. Jain and R. Dubes. *Algorithms for clustering data*. Prentice-Hall, Inc., 1988.

17. A. Jain, M. Murty, and P. Flynn. Data clustering: A review. *ACM Computing Survey*, 31(3):264–323, 1999.

18. F. Jensen. *An Introduction to Bayesian Networks*. UCL Press, 1996.

19. K. Karypis, E. H. Han, and V. Kumar. Chameleon: A hierarchical clustering algorithm using dynamic modeling. *IEEE Computer*, 1999.

20. I. Koch. Enumerating all connected maximal common subgraphs in two graphs. *Theoretical Computer Science*, 250(1-2):1–30, 2001.

21. T. Kohonen. Self-organising maps. *Information Sciences*, 1995.

22. J. MacQueen. Some methods for classification and analysis of multivariate observations. In *Proceedings of the fifth Berkeley symposium on mathematical statistics and probability*, volume 1, 1967.

23. G. McLachlan and T. Krishnan. *The EM algorithm and extensions*. Wiley series in probability and statistics. Wiley, 2. ed edition, 2008.

24. J. Moon and L. Moser. On cliques in graphs. *Israel Journal of Mathematics*, 3(1):23–28, 1965.

25. M. E. J. Newman. The structure of scientific collaboration networks. *Proceedings of the National Academy of Sciences of the United States of America*, 98(2):404–409, 2001.

26. M. E. J. Newman and M. Girvan. Finding and evaluating community structure in networks. *Physical Review E*, 69(2), 2004.

27. G. Palla, I. Derenyi, I. Farkas, and T. Vicsek. Uncovering the overlapping community structure of complex networks in nature and society. *Nature*, 435(7043):814–818, 2005.

28. J. Peña, J. Lozano, and P. Larrañaga. Learning recursive bayesian multinets for data clustering by means of constructive induction. *Mach. Learn.*, 47, 2002.

29. M. Sato, Y. Sato, and L. C. Jain. *Fuzzy Clustering Models and Applications*. Physica-Verlag, 1997.

30. G. Sheikholeslami, S. Chatterjee, and A. Zhang. Wavecluster: A multi-resolution clustering approach for very large spatial databases. In *VLDB'98, Proceedings of 24rd International Conference on Very Large Data Bases*, pages 428–439. Morgan Kaufmann, 1998.

31. T. Zhang, R. Ramakrishnan, and M. Livny. BIRCH: an efficient data clustering method for very large databases. In *SIGMOD '96: Proceedings of the 1996 ACM SIGMOD International Conference on Management of Data*, pages 103–114. ACM Press, 1996.

32. B. Zhou, D. Cheung, and B. Kao. A fast algorithm for density-based clustering in large database. In *PAKDD*, volume 1574 of *Lecture Notes in Computer Science*, pages 338–349. Springer, 1999.