Cautious relational clustering: A thresholding approach

Marie-Hélène Masson, Benjamin Quost, Sébastien Destercke

Cautious relational clustering: a thresholding approach

Marie-Hélène Masson[a,b], Benjamin Quost[b,*], Sébastien Destercke[b]

[a] *Université de Picardie Jules Verne, France*
[b] *Sorbonne Universités, Université Technologique de Compiègne, CNRS, UMR 7253 - Heudiasyc, 57 Avenue de Landshut, Compiègne, France*

Abstract

We propose in this article a new relational clustering method that can return a partial answer (i.e., a set of clusterings) in some cases. Starting from relational or similarity data, we determine a partial equivalence relation defined on the set of objects (two objects are linked if they belong to the same cluster): the key idea is to allow the method to abstain on some pairwise links because they can not be determined with enough certainty from the data. This cautious equivalence relation represents a set of possible hard clusterings which can be obtained by completing the partial relation. This formalization makes it possible to easily detect ambiguous links and to identify subsets of objects with uncertain relationship. We illustrate the potential interest of our approach as a tool of exploratory data analysis using synthetic and real data sets.

Keywords: Partial clustering, relational data, reliable inference.

1. Introduction

Clustering is a challenging issue in machine learning and expert systems (Ünlü and Xanthopoulos, 2019) which consists in grouping objects of similar kind into categories. The literature distinguishes between partitioning and hierarchical approaches. Hierarchical methods provide a sequence of nested clusters, whereas hard partitioning methods determine a division of the set of objects into non-overlapping subsets such that each data object belongs exactly to one subset. Computing a hard partition of the instances is sometimes difficult; therefore, a number of works have proposed to compute *soft partitions*: notable examples include probabilistic partitions (Biernacki et al., 2000), fuzzy partitions (De Oliveira and Pedrycz, 2007; Zhu and Xu, 2018), and credal partitions (Masson and Denoeux, 2008).

*Corresponding author
Email addresses: mylene.masson@hds.utc.fr (Marie-Hélène Masson), benjamin.quost@hds.utc.fr (Benjamin Quost), sebastien.destercke@hds.utc.fr (Sébastien Destercke)

Methods differ also in the kind of data used to determine the clustering. In the case of *object data*, the input is an explicit description of the individuals in the form of a design matrix X containing the descriptions $\boldsymbol{x}_1, \ldots, \boldsymbol{x}_n$ of the n individuals according to a set of descriptive variables, and from which a (geometrical or statistical) model is generally to be estimated. In relational approaches, the clustering is to be determined from a *relational matrix* containing pairwise similarities, or scores, between the n individuals, which are in this case not explicitly described. This latter approach can be considered as more general than the former, since a distance matrix can always be derived from a design matrix, while the converse is generally not possible. The numerical elements of the relational matrix (whether they are similarities, probabilities, ...) can model relations of very different natures: influences in social networks (Carrington et al., 2005), geographic and economical data (Fagiolo and Mastrorillo, 2013), omics data of all kinds (Ben-Dor et al., 1999). Note that such relations do not have to be symmetrical: for example, migration flux between countries or influences in social networks are not.

In this paper, we are interested in proposing a new hard partitioning method dedicated to relational data. Let us recall that, formally, clustering instances amounts to determining equivalence classes, which is formally equivalent to provide the adjacency matrix of the equivalence relation R. This matrix is of general binary term $R_{i,j}$ (with $i, j = 1, \ldots, n$) which indicates the presence ($R_{i,j} = 1$) or absence ($R_{i,j} = 0$) of a relation between a pair of objects $\boldsymbol{x}_i$ and $\boldsymbol{x}_j$. Matrix R actually represents an equivalence relation iff it satisfies the following properties: it must be reflexive (with diagonal terms $R_{i,i} = 1$, for $i = 1, \ldots, n$), symmetric ($R_{i,j} = R_{j,i}$, for $i, j = 1, \ldots, n$) and satisfy the transitivity constraint (if $R_{i,j} = R_{j,k} = 1$, then $R_{i,k} = 1$ as well). If these conditions are met, R can be put in the form of a *diagonal block matrix* by re-arranging its rows and columns, each block on the diagonal then corresponds to one of the clusters of the partition.

In the spirit of recent research focused on cautious classification or ranking (Yang et al., 2017; Cheng et al., 2012), we introduce in this article a new form of *cautious* clustering. The originality of our approach lies in the fact that in the corresponding output relation matrix R, we can abstain to determine some relations $R_{i,j}$ when information about those are too uncertain, that is we left the value $R_{i,j}$ unknown. The idea is to provide partial yet more reliable information, which can eventually be completed by additional information (possibly provided by an expert). The final result is a cautious, partial relation representing a set of possible hard clusterings. We think our approach has several interesting aspects, such as the following ones:

- as already said, relational approaches can handle both object and relational data, and therefore are quite versatile;
- we do not need to specify on advance the number k of clusters to determine, and on the contrary allow in principle the user to choose between multiple solutions;

- we provide a partial answer without adding complex uncertainty models on top of the binary relation, therefore facilitating the result reading compared to probabilistic, fuzzy or evidential relational approaches (Long et al., 2007; Denoeux and Masson, 2004; Masson and Denoeux, 2009).

To our knowledge, there are no proposals in the literature whose goal is to make the relational matrix imprecise, so as to keep only those most reliable links. There are however approaches that can result in other kinds of imprecise clusterings Masson and Denoeux (2008); Lingras (2001), and we detail the interest of choosing the relational view as an alternative representation in Section 2.3.

In a nutshell, our approach to obtain an incomplete relation from an input score matrix consists in thresholding the input relational data, up to the point where it becomes consistent with a hard clustering and beyond. The paper is organized as follows. Section 2 describes the setting, and briefly discusses the interest of focusing on relational information rather than on a design matrix. Section 3 explains how a score matrix S accepted as input can be transformed into a partial equivalence relation via simple operations, in order to obtain a partial clustering of the instances; it also discusses the problem of enumerating the possible (hard and complete) clusterings which can be derived from a partial clustering of the instances. Section 4 presents some preliminary experiments on various synthetic and real clustering problems, so as to illustrate how scoring matrices can be obtained and highlight the potential interest of our approach. Eventually, Section 5 concludes the paper.

2. Setting

In this section, we start by introducing notations used in the remaining of the paper, before introducing the problem addressed.

2.1. Notations

We consider a set $\mathcal{X} = \{\boldsymbol{x}_1, \ldots, \boldsymbol{x}_n\}$ of objects, of which we want to compute a partition $P = (\omega_1, \ldots, \omega_K) \subseteq \mathcal{X}$; that is, $\bigcup_k \omega_k = \mathcal{X}$ and $\omega_k \cap \omega_\ell = \emptyset$ for any $k \neq \ell$. We will denote by $\mathcal{P}$ the set of possible partitions P. In order to simplify notations, and since our approach is not based upon any geometrical or statistical model, we will refer to the objects by their index.

As said in the introduction, a partition is formally the same as specifying an adjacency matrix R of the corresponding equivalence relation, that is encored $R_{i,j} = 1$ if $i \,\mathrm{R}\, j$, and $R_{i,j} = 0$ if $i \neg \mathrm{R}\, j$. Being an equivalence relation, this matrix should satisfy the following conditions:

- it should be symmetric ($R_{i,j} = R_{j,i}$, for all $i, j \in \{1, \ldots, n\}$),
- it should be reflexive, that is its diagonal elements should be non-zero ($R_{i,i} = 1$, for all $i \in \{1, \ldots, n\}$);
- it should be transitive, that is for all $i, j, k \in \{1, \ldots, n\}$ we should have $R_{i,j} = R_{j,k} = 1 \Rightarrow R_{i,k} = 1$, for any $i, j, k \in \{1, \ldots, n\}$.

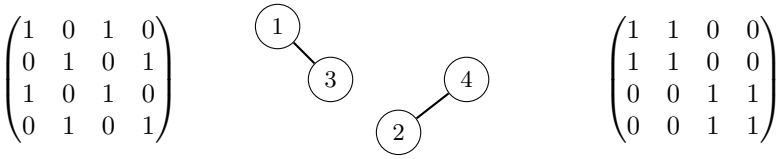


Figure 1: Adjacency matrix and associated graph; derived block matrix.

We will denote by $R_{A,B}$ the sub-matrix consisting of the lines $i \in A \subseteq \{1,\ldots,n\}$ and columns $j \in B \subseteq \{1,\ldots,n\}$. Equivalently, a clustering can be represented by a undirected graph $G = (V, E)$, where the vertices V correspond to the objects and the edges are such that $(i,j) \in E$ iff $R_{i,j} = 1$.

Example 1. Consider the set of individuals $\{1,2,3,4\}$ partitioned into two clusters $\omega_1 = \{1,3\}$ and $\omega_2 = \{2,4\}$. Figure 1 displays the corresponding matrix R and graph G. The matrix R is block: its second and third lines and columns can be switched in order to write it as a block-diagonal matrix.

2.2. Inconsistent and partial matrices

Let us now consider a adjacency matrix corresponding to a general relation R (not necessarily an equivalence one). If such a matrix does not satisfy the 3 properties of an equivalence relation (symmetry, reflexivity, transitivity), then it will be referred to as *inconsistent*. Note that, algorithmically, checking those properties is rather easy: symmetry and reflexivity are straightforward, and transitivity simply amounts to check that $R = R^2$.

In this paper, we will deal with non-fully specified adjacency matrices. A matrix R is complete if $R_{i,j} \in \{0,1\}$ for all $i = 1,\ldots,n$, $j = 1,\ldots,n$. Whenever some elements $R_{i,j}$ are unknown (which will be written $R_{i,j} = ?$), the matrix will be referred to as *partial*. A partial matrix is *consistent* if each of its unknown elements can be replaced by either 0 or 1 so that the resulting complete matrix is consistent; it will be qualified as *inconsistent* otherwise (i.e., missing elements cannot be replaced so that the resulting complete matrix can be rearranged into a block-diagonal matrix). Since $R_{i,i} = 1$ for all $i = 1,\ldots,n$, inconsistency arises whenever either transitivity or symmetry is violated. We remark here that a partial matrix corresponds to a partially specified graph (hereafter referred to as *partial graph*), where $(k,l) \in E$ if $R_{k,l} = 1$, $(k,l) \notin E$ if $R_{k,l} = 0$, and we do not know whether $(k,l) \in E$ or $(k,l) \notin E$ if $R_{k,l} = ?$. We will represent such missing edges by dotted lines throughout the paper.

Example 2. Figure 2 provides the illustration of an adjacency matrix for a set $O = \{\boldsymbol{x}_1,\ldots,\boldsymbol{x}_4\}$ of four objects. It violates symmetry (we have $R_{2,1} \neq R_{1,2}$) as well as transitivity (while $R_{2,4} = R_{4,2}$ satisfy symmetry, we have $R_{4,2} = R_{2,1} = 1$ but $R_{4,1} = 0$). It is relaxed into a partial consistent relation matrix shown in Figure 2.

The various completions provided by the figures correspond to the partitions

$$P_1 = \Big\{\{1\},\{2,3\},\{4\}\Big\}, \quad P_2 = \Big\{\{1,2,3\},\{4\}\Big\}, \quad P_3 = \Big\{\{1\},\{2,3,4\}\Big\}.$$

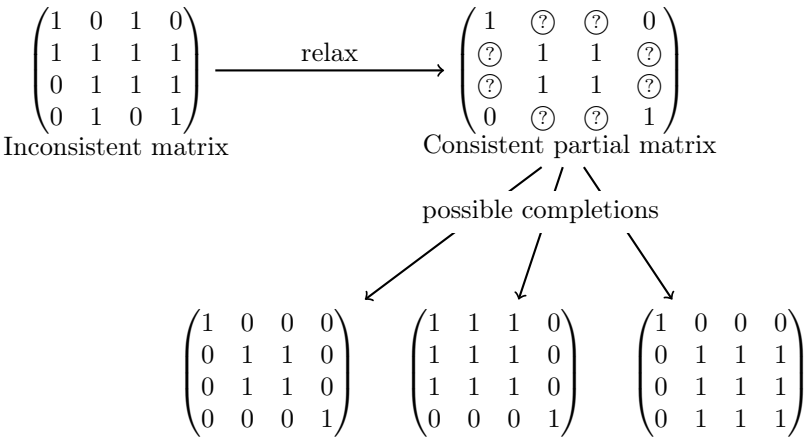


Figure 2: An inconsistent matrix, and a possible relaxed consistent partial matrix with its completions.

2.3. Partial assignments vs partial relations

The idea of providing partial clusterings is not new to the literature. The closest idea to ours that we have found in the literature consists in providing upper and lower approximations of clusters, originally proposed in the rough set field (Lingras, 2001). This idea consists in specifying, for each cluster ω_k, a subset of objects $\underline{\omega}_k$ that necessarily belongs to it, and a subset of objects (including the first) $\overline{\omega}_k$ that potentially belong to it, which only informs us that $\underline{\omega}_k \subseteq \omega_k \subseteq \overline{\omega}_k$. Alternatively, given K clusters $\omega_1, \ldots, \omega_K$, partially clustering the instances may also be seen as identifying the set of plausible clusters for each instance $\boldsymbol{x}_i$, i.e. to predict a subset $\kappa_i \subseteq \{\omega_1, \ldots, \omega_K\}$ of clusters to which it may belong. This is essentially the strategy behind approaches providing a soft partition of a set of instances (Masson and Denoeux, 2008).

However, such approaches to get partial clusterings are not equivalent to the one explored in this paper. First, it should be stressed out that determining the plausible clusters for a set of instances generally requires to specify the number of clusters to consider. When processing relational data, the information at hand does not make any assumption regarding this number, which can instead be inferred from the relation matrix obtained as output.

As an illustration, consider again the clustering problem in Example 2, and let κ_i stand for the set of possible clusters to be determined for the ith individual. The consistent partial matrix gives three pieces of information: (a) 2 and 3 should be together; (b) they may either be in the same cluster as 1, or 4; (c) 1 and 4 must be in two separate clusters.

Let us first assume that we are looking for a partition into $K = 2$ clusters. The only way to satisfy constraint (c) is then to assign 1 and 4 to single clusters: for instance, $\kappa_1 = \{\omega_1\}$ and $\kappa_4 = \{\omega_2\}$. However, satisfying both (a) and (b) amounts to set $\kappa_2 = \kappa_3 = \{\omega_1, \omega_2\}$: then, the information that 2 and 3 should

remain in the same cluster has been lost. In the case of a partition into $K = 3$ clusters, satisfying (a) leads to $\kappa_2 = \kappa_3 = \{\omega_2\}$; (b) translates into $\kappa_1 = \{\omega_1, \omega_2\}$ and $\kappa_4 = \{\omega_2, \omega_3\}$, which violates (c): even if objects 1 and 4 *could be* related to (2,3), they *cannot be at the same time*.

3. Processing score matrices

In this work, we assume our input information to be a $n \times n$ matrix S of graded scores $S_{i,j}$. We interpret $S_{i,j}$ as an information about the relation $R_{i,j}$ between two objects. We assume that these scores are within an interval $[a, b]$: the closer a score $S_{i,j}$ is to a (respectively, b), the higher is the belief that objects i and j are disconnected (resp., related) to each other. In addition, we will assume a "neutral" element $c \in [a, b]$, according to which uncertain relations can be identified: in other words, scores $S_{i,j} \in [a, c)$ (respectively, $\in [c, b]$) support the conclusion $R_{i,j} = 0$ (resp., $R_{i,j} = 1$), and this support is considered to be weak if the score is close to c. Apart from this, we do not assume that scores $S_{i,j}$ satisfy any specific property in general. In practice, we will often choose $c = {}^{(a+b)}/_2$. For example, probabilistic predictors would provide a score $S_{i,j} \in [0, 1]$, with 0.5 as the neutral element; other techniques may give real-valued scores $S_{i,j} \in (-\infty, +\infty)$, with 0 acting as a neutral element. Note that this neutral element may alternatively be computed from the data so as to exhibit specific properties (such as, e.g., the median in a set of scores), or specified by a user.

Obviously, we can easily transform S into a binary matrix R by setting $R_{i,j} = 0$ if $S_{i,j} < c$, and $R_{i,j} = 1$ otherwise. However, the resulting matrix R will likely be inconsistent if the scores $S_{i,j}$ are estimated independently from each other. We thus propose to consider partial adjacency matrices, whose partiality is induced by the scores $S_{i,j}$ and by how close they are to c.

We thus propose to distinguish between scores according to their degree of support in favor of a relation or absence of relation. To this end, we propose to define a partial matrix R^ε such that $R^\varepsilon_{i,j} = 0$ if $S_{i,j} < c - \varepsilon$, $R^\varepsilon_{i,j} = 1$ if $S_{i,j} \geq c + \varepsilon$, and $R^\varepsilon_{i,j} = ?$ otherwise, for some $\varepsilon \geq 0$. This amounts to assess that $R^\varepsilon_{i,j} = ?$ whenever $c \in [S_{i,j} - \varepsilon, S_{i,j} + \varepsilon] \cap [a, b]$. The parameter ε obviously plays a crucial role in our procedure, since its value will impact the number of missing relations in the partial matrix obtained.

Example 3. Figure 3 displays a score matrix ($S_{i,j} \in [0; 1]$ for all $i, j \in \{1, \ldots, n\}$); thresholding this matrix with respect to the typical neutral element $c = 0.5$ leads to an inconsistent relation matrix R: it can be easily checked that both symmetry and transitivity are violated.

Discounting the matrix using a value $\varepsilon = 0.1$ leads to a discounted (interval-valued) score matrix. An interval should be interpreted as the set of plausible values for the actual score $S_{i,j}$: it can thus be considered as symmetric whenever two intervals corresponding to the same score have a nonempty intersection.

The discounted score matrix can in turn be thresholded using the same neutral element $c = 0.5$: the resulting relation matrix is now partial but consistent. Note that it is the same as the one obtained in Example 2.

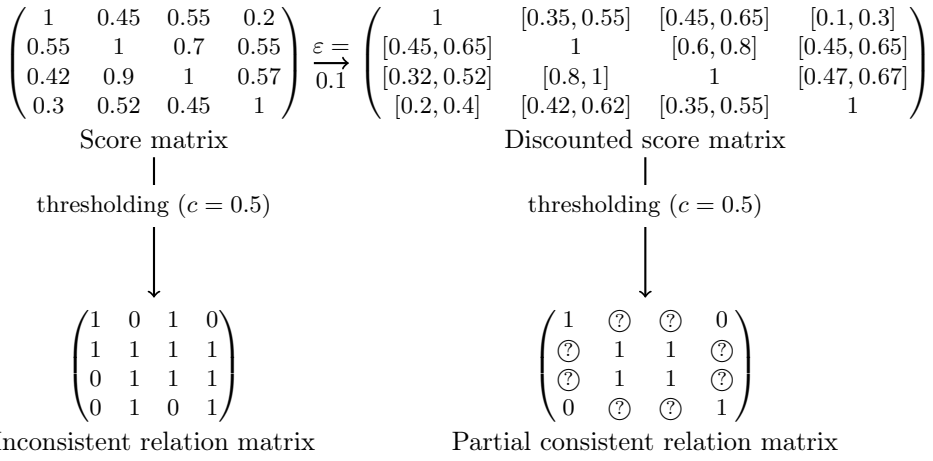


Figure 3: Input score matrix S with associated inconsistent relational matrix R; discounted score matrix ($\varepsilon = 0.1$) leading to a partial consistent matrix R^{ε}.

Our strategy consists first in computing the smallest value of ε resulting in a consistent matrix R^{ε}, so as to ensure that the original scores $S_{i,j}$ are altered as little as possible in the process. The minimal value satisfying this property will hereafter be written ε^*, and R^* will stand for the corresponding partial consistent relational matrix. Computing ε^* and R^* is one of the main algorithmic concerns of this paper. Note that we may also imagine that imprecise scores are available from the start, and our approach is versatile in this respect, since the methods can easily be applied to these cases (or to combinations of them).

3.1. Checking and obtaining consistency

As explained above, the main concern of this paper is to compute a minimal value ε^* leading to a partial but consistent matrix R^*. A consistent relational matrix satisfies reflexivity, symmetry and transitivity. As suggested previously, checking for the two former is straightforward; for the latter, our strategy relies on the fact that the graph $G = (V, E)$ corresponding to a proper partition is a set of disjoint cliques: that is, every connected component D_i is fully connected[1].

Proposition 1. *A partial matrix R is consistent if and only if every connected component D_i of the corresponding graph can be completed into a fully connected graph (a clique), i.e. $0 \notin R_{D_i,D_i}$.*

Proof. **If**: Assume that $0 \notin R_{D_i,D_i}$ for any connected component. First note that we cannot have $R_{k,l} = 1$ if vertices k and l are not in the same connected

[1]Recall that a subset $D_i \subseteq V$ of nodes is a connected component if there exists a path between each pair of nodes $k, l \in D_i$; it is fully connected if $(k, l) \in E$ for all $k, l \in D_i$. The same holds for partial graphs, where (fully) connected components are identified from known links.

component. Therefore, for any $k \in D_i, l \in D_j$ with $j \neq i$, we have $R_{k,l} \in \{?, 0\}$. Since the set of connected components forms a partition (i.e. $D_i \cap D_j = \emptyset$ for all $i \neq j$, and $\cup_i D_i = V$), we can always take the following completion for any $R_{k,l} = ?, k \in D_i, l \in D_j$:

$$R_{k,l} = \begin{cases} 1 & \text{if } i = j \\ 0 & \text{else} \end{cases} \tag{1}$$

That is, if each pair of connected components remains unconnected and forms a clique, then the matrix is consistent as it has at least one completion into a partition.

Only if: assume we have $R_{k,l} = 0 \in R_{D_i,D_i}$ for some $k, l \in D_i$. Since D_i is a connected component, then there is a path from vertex k to vertex l, and by transitivity we should have $R_{k,l} = 1$, showing that R is not consistent. □

Algorithm 1 describes a simple method for checking consistency, derived from Proposition 1; it makes use of any existing efficient method to extract connected components (see, e.g., (He et al., 2017)).

Algorithm 1: Check consistency of R

Input: partial matrix R
Output: assessment Cons (Cons=0: inconsistent, Cons=1: consistent)
Cons=1;
Extract connected components $D_1, \ldots, D_L$;
foreach component D_i **do**
  if $\exists k, \ell \in D_i^2$ with $R_{k,\ell} = 0$ **then** set Cons=0 and stop;
return Cons

Then, the optimal value ε^* can be obtained either by starting with $\varepsilon = 0$ and increasing its value until R^ε is consistent, or by a dichotomic search between 0 and $\overline{\varepsilon}$ (with $\overline{\varepsilon}$ necessarily leading to a partial consistent relation matrix, typically $\overline{\varepsilon} = \max_{i,j} |R_{i,j} - c|$). Algorithm 2 describes this latter procedure: a lower bound ε_* and an upper bound ε^* on ε are updated so as to converge to each other (with ε^* always leads to a partial consistent matrix), and the procedure stops when the current value ε is in a δ-neighborhood of ε^*. The parameter δ is chosen by the user.

3.2. Deducing values in partial consistent matrices

Given a generic consistent partial matrix R, it may be possible to deduce some missing values by exploiting the symmetry and transitivity properties. For instance, if $R_{i,j}$ is known and $R_{j,i} = ?$, we can immediately deduce the value of the latter by symmetry — the same result may be achieved by exploiting transitivity, for instance if $R_{i,j} = 1$ and $R_{i,k} = 1$ while $R_{j,k} = ?$.

Algorithm 2: Obtain minimal discounting parameter ε^*

Input: score matrix S, precision δ, $\overline{\varepsilon}$
Output: ε^*
$\varepsilon_* \leftarrow 0$;
$\varepsilon^* \leftarrow \overline{\varepsilon}$;
$\varepsilon \leftarrow {}^{(\varepsilon_* + \varepsilon^*)}/_2$;
while $|\varepsilon^* - \varepsilon| \geq \delta$ **do**
  Cons(ε) $\leftarrow$ output of Algorithm 1 for R^ε ;
  if Cons(ε) $= 1$ **then** $\varepsilon^* \leftarrow \varepsilon$ **else** $\varepsilon_* \leftarrow \varepsilon$;
  $\varepsilon \leftarrow {}^{(\varepsilon_* + \varepsilon^*)}/_2$;

In this section, we show how such missing relations can be identified from a given consistent partial matrix R. Once these *necessary* replacements[2] are made, the remaining missing relations $R_{i,j} = ?$ are impossible to deduce without adding further information: they could in principle be replaced either by 1 or 0 (provided that such *unnecessary* replacements are consistent with each other).

Our approach to proceeding with necessary replacements in a partial matrix R consists in two successive steps:

- for each connected component D_i of the graph given by R (obtained by Algorithm 1), we have $R_{D_i,D_i} = \mathbf{1}_{|D_i|}$, which follows from transitivity: this means that any $? \in R_{D_i,D_i}$ must be replaced by 1.
- Then, if for any pair of distinct components D_i, D_j with $i \neq j$, we observe $0 \in R_{D_i,D_j}$ (and thus $0 \in R_{D_j,D_i}$ due to symmetry), R cannot be completed so that $R_{D_i \cup D_j, D_i \cup D_j}$ forms a unit matrix (i.e., components D_i and D_j cannot be aggregated into a fully connected component): therefore, all remaining missing elements $?$ must be replaced by 0.

Once the partial matrix has been processed through these two steps, all remaining pairs of disjoint connected components D_i, D_j such that R_{D_i,D_j} and R_{D_j,D_i} contain missing elements can either be linked or separated, by replacing the missing relations between their elements either with ones or zeros.

An illustration of this procedure is provided in Figure 4, where we identify first all mandatory relations ($?$ in R to be replaced by 1), and then all mandatory separations ($?$ in R to be replaced by 0). The end result is still imprecise, as it still contains some $?$ elements. It is clear that the result of our procedure corresponds to a unique clustering if and only if the completed matrix is consistent and no longer contains any $?$ elements. In such a case, our procedure may be seen as "repairing" the initial inconsistent matrix.

[2]Necessary replacements are constrained by identified relations in the matrix: the missing value must be uncovered using the available information.

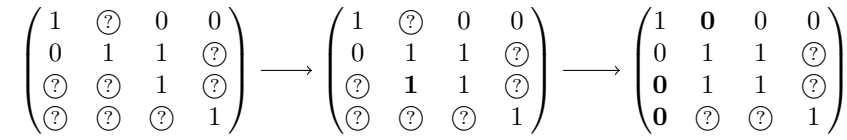


Figure 4: Deducing matrix elements.

3.3. Computing completions: a discussion

A partial clustering such as provided by the procedure described in Section 3.2 can deliver cautious inference to the user. Indeed, it makes it possible to identify relations strongly supported by data, and to abstain to predict on those where we have insufficient or ambiguous information. Such abstention could provide a basis for performing active learning of clustering relationships.

In this part, we discuss the problem of obtaining completed matrices from partial consistent ones. We will keep the discussion rather general, as our primary goal in this work is to keep partiality in order to exploit it, and not especially to suppress it in some preprocessing. Nevertheless, having methods such as the ones presented here to scan among completions of partial matrices could be useful, e.g., to present some of them to users.

Guessing the number of clusters. In some contexts, it is natural to estimate how many different completions could be made and how many clusters could result from such completions, since our approach does not require to fix the number K of clusters in advance. This latter question can be addressed by providing lower and upper bounds $(\underline{K}, \overline{K})$ to the number of clusters.

Determining an exact upper bound is actually easy, since replacing every ⑦ by zeros separates completely connected components from each other whenever possible: as a consequence, if $D_1, \ldots, D_L$ are the connected components of R, then $\overline{K} = L$.

Determining a lower bound $\underline{K}$, however, cannot be achieved by simply replacing missing values with 1, since it may result in an inconsistent matrix. To determine $\underline{K}$, let us consider the graph $G' = (V, F)$ encoding the identified absences of relations, where the vertices are the objects and $(i, j) \in F$ iff $R_{i,j} = 0$. It is clear that each cluster corresponds to a set of objects which are not connected in G', and that any pair of connected objects should belong to different clusters in any possible completion of G'. The minimal amount of clusters satisfying the identified absences of relations can be obtained by solving the coloring problem in G', and by taking $\underline{K} = C$, with C the number of colors required so that every connected vertices of G' have different colors.

Enumerating the set of full relation matrices consistent with the data. A more complex issue is that of computing the set of possible completions of a partial consistent matrix R obtained from the data, as was done in Example 2. A simple procedure for determining this set consists in using a recursive strategy

such as described by Algorithm 3: missing values are progressively replaced by 1 and then 0, before the matrix is completed and the procedure called again, stopping when a full matrix is obtained.

Algorithm 3: Compute the set $\mathcal{R}$ of full relation matrices consistent with R

Input: relation matrix R
Output: set $\mathcal{R}$ of complete consistent matrices
if R *is complete* **then**
  return $\mathcal{R} \leftarrow R$
else
  Pick up a missing relation $R_{i,j}$;
  Define R^{+1} by replacing $R_{i,j}$ by 1, and complete it (see Section 3.2);
  Call recursively Algorithm 3 on R^{+1} and add the output to $\mathcal{R}$;
  Define R^{+0} by replacing $R_{i,j}$ by 0, and complete it (see Section 3.2);
  Call recursively Algorithm 3 on R^{+0} and add the output to $\mathcal{R}$;
  return $\mathcal{R}$

If the amount of missing relations is judged too important for Algorithm 3 to be used, the random strategy described in Algorithm 4 can be used alternatively. It is basically a Monte-Carlo approach for retrieving a set of desired size of full consistent relation matrices from R: the strategy repeats replacing missing relations at random by 0/1 numbers and completing the matrix.

Note that Algorithm 3 makes it possible to retrieve the full set $\mathcal{R}$ of matrices consistent with R by considering all replacements of missing values in this latter. The size of this set is obviously bounded above by $2^{|?|/2}$, where $|?|$ is the number of missing elements — this will usually be a conservative upper bound, since additional missing relations can be deduced using transitivity each time a replacement is made. Algorithm 4 is in principle computationally less expensive, since it only computes a subset $\widehat{\mathcal{R}}$ of desired cardinality T of full matrices consistent with R — although there is a chance of sampling the same matrix several times, hence delaying the computation of $\widehat{\mathcal{R}}$. The probability p of replacing missing values by 1 will directly influence the average number of clusters, a high p being likely to induce a low number of clusters and conversely — in particular, a value $p = 0$ will lead to systematically separating the groups already identified, thus retrieving the solution with $K = \overline{K}$ clusters. Note that any value $p \in (0, 1)$ will, in theory, finish by sampling every possible completions, yet it is unclear in practice how fast it will obtain the full set $\mathcal{R}$.

Computing a clustering using additional information. Eventually, we may consider the problem of determining a specific completion satisfying some properties. There are several ways to reach this objective, such as fixing the number K^* of desired clusters, or choosing the completion which minimizes some discrepancy between the hard partition matrix R and the score matrix S.

Algorithm 4: Compute a subset $\widehat{\mathcal{R}}$ of full relation matrices consistent with R

Input: relation matrix R, desired cardinality T for $\widehat{\mathcal{R}}$, probability p of drawing ones, maximal number N of iterations
Output: set $\widehat{\mathcal{R}}$ of complete consistent matrices deriving from R

```
R̂ ← ∅;
n ← 0;
while |R̂| < T and n < N do
    n ← n + 1;
    R̂ ← R;
    while R̂ contains missing values do
        Pick up a missing value R̂_ij = ? at random;
        Replace R̂_ij by sampling from a Bernoulli distribution B(p);
        Complete R̂ (see Section 3.2);
    if R̂ ∉ R then
        Set R̂ ← R̂ ∪ R̂;
return R
```

Note, however, that if the purpose is to determine a single hard partition from the data, straightforward approaches (such as, e.g., spectral clustering) may be better suited than our strategy, which aims at letting a user provide side knowledge in a cautious, iterative fashion based on intermediate solutions inferred from the data.

3.4. Choosing ε

From our discussion, it is clear that picking the value ε^* will result in a consistent matrix having a minimal number of missing elements. We can even expect that in a number of cases, deductions of Section 3.2 will make this partial matrix complete. Experiments of Section 4 will confirm this. In those situations, our approach corresponds more to repairing an inconsistent matrix than to providing cautious but reliable partial inferences.

This is why, in practice, one can choose a value of ϵ higher than ε^*, so as to only keep those link predictions that are the most certain. Choosing such a value is then similar to picking a rejection threshold or cost in classification (Bartlett and Wegkamp, 2008), and highly depends on how cautious the user is ready to be in order to ensure inference reliability.

It should be noted that, in practice, not all values of ϵ will provide different examples, and there will be intervals of values that will result in the same partial consistent matrix, and there will be a finite number of such matrices. An obvious idea is then to display those different matrices of increasing partiality (choosing a representative ϵ for each of them), and let the user analyse them. This is the strategy we will follow in the experiments.

4. Experiments on clustering problems

This section shows various experiments whose main purposes are to show how our mehod can be used, and to confirm that the induced partiality indeed keep the reliable predictions and forget the least certain ones.

The first two experiments, respectively on simulated and real data sets where data are interval-valued (and hence distances imprecise), show how our method can be applied to object data, confirming its generality. The third experiment provides results on two relational data sets for which the number of sought clusters is unknown, and for which there is no information about the objects themselves. Finally, the last experiment demonstrates that our approach performs as expected, in the sense that those forgotten links are those that degrade the most the clustering quality.

4.1. Synthetic Gaussian data

As a first illustration of our approach, we created a synthetic two-dimensional data set from a bivariate Gaussian distribution with $g = 4$ components, each of them counting $n_k = 25$ instances, with common covariance matrix $\Sigma_k = 1/4\ \mathrm{Id}_2$ (hereafter, Id_p stands for the identity matrix of dimension p) and expectations

$$\mu_1 = \begin{pmatrix} 0 \\ 0 \end{pmatrix}, \quad \mu_2 = \begin{pmatrix} 1.75 \\ 0 \end{pmatrix}, \quad \mu_3 = \begin{pmatrix} 0 \\ 1.75 \end{pmatrix}, \quad \mu_4 = \begin{pmatrix} 1.75 \\ 1.75 \end{pmatrix}.$$

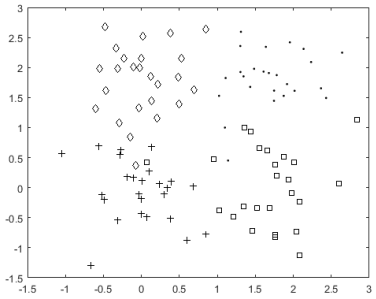
The data are represented in Figure 5a. They are purposely kept very simple to allow for a visualization of the different steps of the method.

The clustering is performed using `MIXMOD`, a software package for model-based supervised and unsupervised classification using mixture models, available at http://www.mixmod.org/. The experiment is conducted in the following way: we sample with replacement in the whole data set; we then estimate the parameters of a Gaussian mixture model and we use them to compute estimates $\widehat{p}_{ij}$ of the co-association probabilities that each pair of instances in the initial data set is in the same cluster:

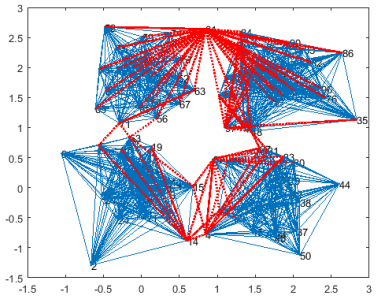
$$\begin{aligned} \widehat{p}_{ij} &= \sum_{k=1}^{K} \widehat{P}(\omega_k \mid \boldsymbol{x}_i)\widehat{P}(\omega_k \mid \boldsymbol{x}_j), \\ \widehat{P}(\omega_k \mid \boldsymbol{x}) &= \frac{\widehat{\pi}_k f(\boldsymbol{x}; \widehat{\theta}_k)}{\sum_{\ell=1}^{K} \widehat{\pi}_\ell f(\boldsymbol{x}; \widehat{\theta}_\ell)}, \end{aligned}$$

where $f(\cdot; \theta_k)$ stands here for the pdf of a multivariate Gaussian distribution with parameters $\theta_k = (\mu_k, \Sigma_k)$ and π_k is the proportion of component k.

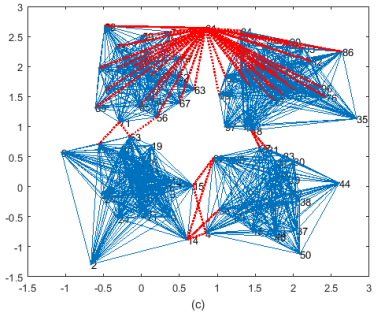
In the experiments, we repeated $n_b = 20$ times the procedure of estimating a mixture of $K = 4$ Gaussians with equal proportions and equal covariance matrices, and averaged the co-association matrices thus obtained. Using a neutral element $c = 0.5$, our strategy provided a minimal discounting value $\varepsilon^* = 0.09$, meaning that the averaged co-association matrix is not consistent (otherwise



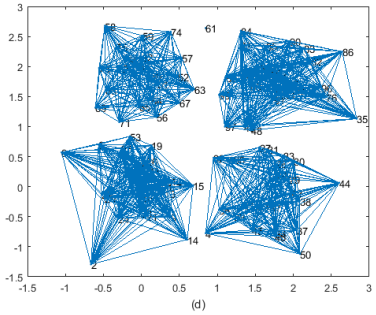
(a) Synthetic Gaussian data



(b) Representation of R^{ε^*}



(c) Completion of R^{ε^*} with ones



(d) Completion of R^{ε^*} with zeros

Figure 5: Illustrative example: synthetic Gaussian data with 4 classes. Blue solid lines represent known associations ($R_{i,j} = 1$) and red dashed lines missing relations (?).

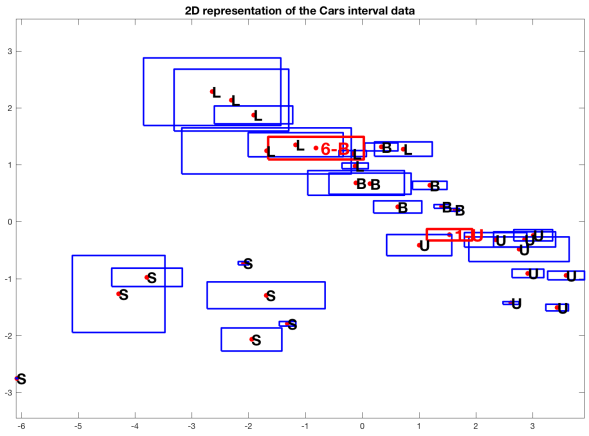


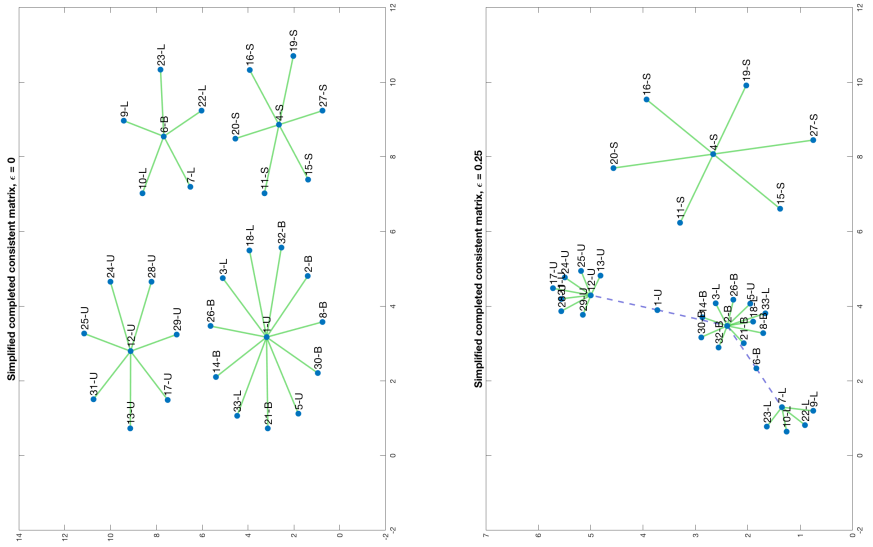
Figure 6: Representation of the Cars data set

we would have obtained $\varepsilon^* = 0$). The representation of R^{ε^*} is given in Figure 5b: blue solid lines represent known associations ($R_{i,j} = 1$ in the partial matrix), and red dashed lines missing relations ($R_{i,j} = ⑦$). The completions with ones and zeros are given in Figure 5c and 5d, respectively. It can be seen that after proceeding with completing the matrix, the four clusters are correctly recovered, except for a point (object n. 61) which remains isolated, since it is never associated with a high confidence with any other instance in any of the n_b clusterings.

This simple synthetic example shows that R^*, resulting from a minimal correction of the initial inconsistent scores, will often be completed into a unique, non-partial clustering: as such, using ε^* can be seen as a "repairing" tool, as said in Section 3.4. For this purpose, we may explore such partial solutions by arbitrarily picking values $\varepsilon \in [\varepsilon^*; 0.5]$: the intuition is that by increasing the discounting level ε, only the most certain pairwise relations will be kept in the resulting matrix R^ε, and thus the most robust information based on which a clustering may be determined. This strategy can be seen as an iterative learning procedure for computing more and more robust clusterings of the data. We may imagine introducing another step where information is actively provided by an user after visualizing the partial clusterings thus obtained.

4.2. Interval-valued data

We now illustrate the approach in the particular case of interval-valued data. We use the Cars data set (Carvalho et al., 2006) which consists in 33 car models described by 8 interval-valued features (price, engine capacity, top speed, acceleration, step, length, width and height). These cars are a priori classified into four categories : Utilitarian, Berlina, Sporting and Luxury.



(a) Clustering with minimal correction (b) Partial clustering obtained with $\varepsilon = 0.25$

Figure 7: Clustering of the Cars data into four clusters.

A 2D representation of the data is obtained using principal component analysis (PCA) for interval-valued data (Cazes et al., 1997). This simple method consists in performing a standard PCA on the centers of the intervals and projecting the rectangles on the axes thus obtained. The projection onto the two first principal axes is displayed in Figure 6. Note that the data have been standardized according to the means and standard deviations of the centers of the intervals.

As previously, our method is applied based on average co-association matrices estimated by resampling in the original data. More specifically, precise samples are repeatedly drawn at random in the initial interval-valued data and then clustered into $K = 4$ clusters using a standard fuzzy C-means algorithm (Bezdek, 1981). This process is repeated $n_b = 20$ times and the co-association matrices obtained from each clustering are averaged. Using a neutral element $c = 0.5$, we find that $\varepsilon^* = 0$ meaning that the averaged co-association matrix is found to be consistent. The partition obtained is represented in Figure 7a. For the sake of clarity, we did not represent the transitive closure of the objects in each cluster as before. Instead, an object was arbitrarily chosen in each cluster and used to represent intra-cluster and inter-cluster relations. The partition is

consistent with the results provided by other authors (see e.g. (Carvalho and Lechevallier, 2009b,a)).

So, in this case, the main interest of our method is to explore values $\varepsilon > \varepsilon^*$, and to see if the made cautious inferences are meaningful. In this case, we only display the least partial clustering, corresponding to the first non-complete matrix R^ε as ε increases. A representative value for this is $\varepsilon = 0.25$: choosing this value and completing the resulting matrix with 1 and 0 gives the imprecise clustering represented in Figure 7b. This partial clustering isolates the instances *6-B* and *1-U* (indicated in red bold in Figure 6), which were previously misclassified, and identifies each of them as a potential member of two clusters, among which is the right one.

4.3. Relational data

We now present some results obtained on two relational datasets. The Mutation data (Fitch and Margoliash, 1967) consist in dissimilarity measures between 20 species. More specifically, it is based on the Cytochrome C protein molecule, which is highly conserved across animals, plants, and many unicellular organisms, and whose structure varies according to the species. Due to its wide spread and small size, it has been used to construct phylogenetic trees, based on the numbers of positions with different acids in the Cytochrome C amino-acid chain for each pair of species. The Airports data consist in geodesic distances between airports located in various countries worldwide. Note that none of these two datasets have any class information.

In both cases, the dissimilarities are first normalized with respect to the highest observed dissimilarity in the data. We then obtain the score matrix by taking, for each pair of elements, the complement to one of the normalized dissimilarity: if d_{ij} is the dissimilarity between elements $\mathbf{x}_i$ and $\mathbf{x}_j$, then we have $S_{i,j} = 1 - d_{ij}$.

Mutation data

A 2D representation of the species described in the Mutation data, obtained by classical multidimensional scaling, is given in Figure 8. The representation in the first factorial plane corresponds to 69.66% of the variance corresponding to the positive eigenvalues.

As it turns out, the score matrix derived from the Mutation dissimilarity data is inconsistent: our procedure gives a minimal value $\varepsilon^* = 0.07$ in order to obtain a partial consistent matrix. The corresponding clustering is given in Figure A.11a (see Appendix A). As can be seen, the minimally relaxed relation matrix can lead to two different clusterings, whether "Bread yeast" and "Skin fungus" species are grouped into the same cluster or not.

We have also computed three different partial clusterings for this problem, for the increasing values $\varepsilon \in \{0.20, 0.25, 0.30\}$ (these are representative values of increasingly partial matrices). All results are illustrated in Figures A.11b to A.11d (Appendix A). The results for $\epsilon = 0.25$ are displayed in larger in Figure 9. For $\varepsilon = 0.20$, the pairwise links between "Tuna", "Screwworm fly, Moth" and

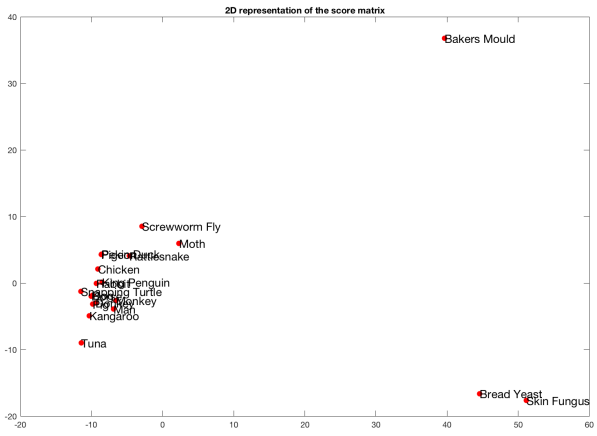


Figure 8: Representation of the Mutation data.

the main cluster have been relaxed into uncertain links, "Screwworm fly, Moth" being still considered as a cluster. Setting $\varepsilon = 0.25$ further leads to consider the link between "Rattlesnake" and the main cluster as uncertain. With $\varepsilon = 0.30$, the link between "Screwworm fly" and "Moth" also becomes uncertain, as is a potential link between "Baker's mould" and "Bread yeast, skin fungus", which was previously considered as impossible.

Airports data

Similarly to the mutation data set, we obtained a 2D representation of the airports obtained by classical multidimensional scaling, in Figure B.12 of Appendix B. This representation corresponds to 72.83% of the variance corresponding to the positive eigenvalues. As previously, the score matrix derived from the Airports dissimilarity data is not consistent and requires at least $\varepsilon^* = 0.27$ in order to obtain a partial consistent matrix. The associated clustering consists in a single solution with three clusters — in a nutshell, Australia, America and Africa-Asia-Europe.

In addition to the clustering obtained with $\varepsilon^* = 0.27$, we also computed three partial clusterings for increasing values $\varepsilon \in \{0.31, 0.32, 0.33\}$ (again, these are representative values of increasingly partial matrices), which are represented in Figures B.13b to B.13d (Appendix Appendix B). For $\varepsilon = 0.31$, the link between "Los Angeles, San Francisco" and the other american airports has been relaxed, as well as the absence of link between the australian and american clusters: the data may thus be clustered into two to four clusters. With $\varepsilon = 0.32$, the link between the Asia and Africa-Europe clusters become uncertain, the former being in addition potentially related to the american (and therefore the australian) airports. The solutions thus range from one to five clusters. Increasing ε to 0.33

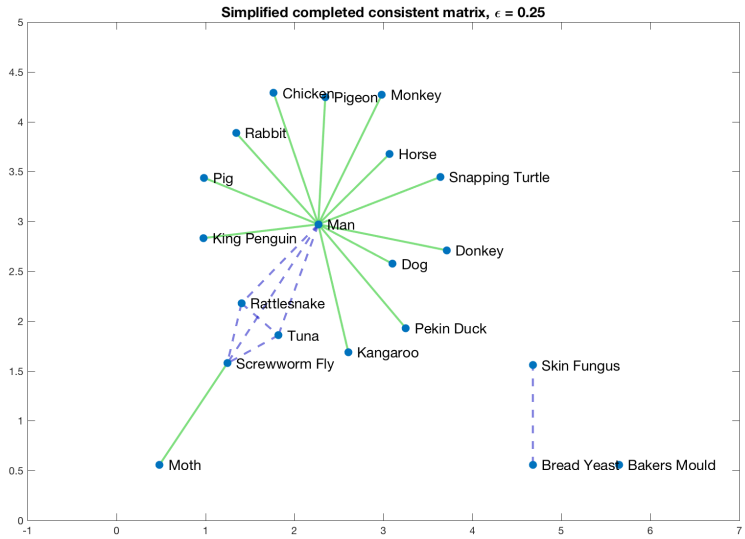


Figure 9: Partial clusterings of the Mutation data for $\varepsilon = 0.25$.

separates "Montreal, New York" from "Caracas, Lima, Buenos Aires", thereby leading to a set of solutions ranging from one to six clusters.

4.4. Completeness-correctness tradeoff

As we have argued before, one interest of our approach is to consider different levels for ϵ, making the predictions more robust as they become more partial. An expected behaviour of our approach is therefore that the accuracy of the remaining links increases as we abstain more, in contrast with simply forgetting at random, which by principle would not increase on average the accuracy of the remaining links. The experiments performed in this section aims at confirming this intuition, and at showing that the expected behaviour is indeed the one observed. We illustrate this fact on six classical datasets from the UCI Machine Learning Repository (Lichman, 2013) described in Table 1.

In these experiments, we used the same process than for the synthetic data set. Bootstrap samples are created by sampling with replacement in the whole data set. A Gaussian mixture model is then fit to each bootstrap sample using `MIXMOD` (with a number of components equal to the known number of classes), and used to compute the probability for each pair of instances in the initial dataset to belong to the same cluster. We repeat this process $n_b = 20$ times and average the co-association matrices.

[3]Three clusters of very small size have been removed from the original data set.

Table 1: Datasets description

data set	#features	#labels	#instances
iris	4	3	150
wine	13	3	178
seeds	7	3	210
ecoli[3]	7	5	327
segment	18	7	2310
optdigits	64	10	3823

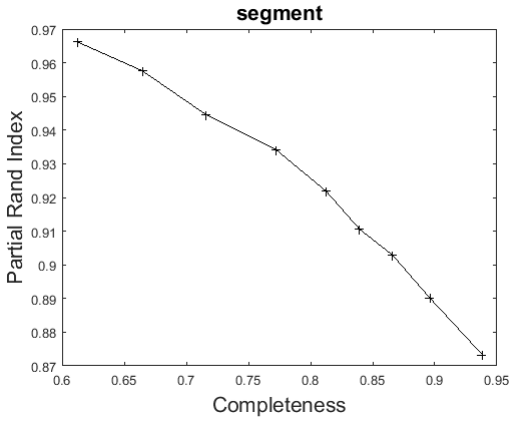


Figure 10: Clustering results on segment

Since our approach potentially yields an incomplete clustering, we use two specific measures to evaluate the results of the experiments. The first one is an extension of the Rand index (Rand, 1971) which is classically used to measure the similarity between two partitions. The Rand index computes the proportion of pairs of instances on which both clusterings agree: either they are classified in the same subset by the two clusterings, or they are classified in different clusters in both clusterings. A natural extension for partial clusterings consists in computing the Rand index only using the pairs (i, j) of objects for which $R^{\varepsilon}_{i,j} \neq ?$. The second criterion, the completeness of the relation, corresponds to the proportion of pairs (i, j) for which $R^{\varepsilon}_{i,j} \neq ?$. A good method for providing partial clusterings should see the Rand index increase when completeness decreases. A tradeoff between completeness and correctness can be reached by setting ε between ε^* and 0.5, thresholding R and completing the relation with ones and zeros.

Figure C.14 in Appendix C displays the average completeness and partial Rand index, computed over ten repetitions of the process described above, for all data sets. Figure 10 provides the result for the segment data set only, whose behaviour is similar to that of other data sets. It turns out that the completeness

is never equal to 1, which means that the initial matrices are never consistent. This also show that while using ε^* may lead to precise consistent matrix, this is not always the case. As expected, abstaining to make a complete clustering in presence of uncertain relational information leads to improved performances. For several datasets, the accuracy can be significantly increased at the price of a reasonable decrease in completeness, as we always reach a Rand index above 0.95, while never going below a completeness of 0.6, meaning that more than half the links remain.

5. Conclusions

In this paper, we have introduced an approach for computing a partial clustering from relational data, more precisely based on a matrix containing scores of relations between pairs of items. A score matrix may not be consistent (i.e. it may not encode a proper equivalence relation): therefore, the first step of our approach consists in (minimally) relaxing it until it satisfies the required symmetry, reflexivity and transitivity properties. The corresponding matrix is then completed by exploiting these properties. If the completed matrix is still incomplete, the solution is considered to be imprecise, in that several clusterings can be derived from the data. The main interest of our approach lies in the fact that in presence of scarce information, it allows for drawing cautious conclusions from the data, possibly until additional information is provided by an expert.

In the experiments, we show how score matrices can be generated, either by probabilistic generative models such as mixture of Gaussian distributions, or by sampling in imprecise data. In both cases, the results point out that allowing for partial clusterings can be helpful to identify ambiguous items (i.e., which could belong to several clusters), and increase our confidence in the results while still providing meaningful clustering outputs. Experiments realized on real relational datasets show the interest of our approach in an exploratory data analysis process, since the user can easily be provided with a feedback on possible solutions induced by the data.

To our knowledge, our proposal is the first to investigate the possibility to provide partial clusterings in a relational manner. A closely related, yet different problem is the detection of outliers (Melendez-Melendez et al., 2019; Tellaroli et al., 2016) in the clustering problem, that can abstain to assign some object to the clusters. It therefore corresponds to a reject strategy (objects are either assigned to one cluster, or not at all), a specific case of a partial assignment. It would be interesting to investigate in the future how our approach can help to make such a detection. Similarly, it would be interesting to investigate in which measure our approach can help in other connected problems such as in active learning (Rendle and Schmidt-Thieme, 2008), where it is essential to detect which information si unreliable, and should be asked to the experts, or in signed graph problems (Figueiredo and Frota, 2014), where the goal is to find a bi-partition despite of noisy and uncertain information about the links between individuals (typically voters forming coalitions).

There also remain some limiting aspects of our approach that could be improved. For instance, while the described methods remain tractable even in presence of a large amount of items, as it involves only polynomial algorithms, a full visualisation of the results as we gave in the experiments would be infeasible. If we were for example tackling active learning problems for such large data sets, it would be necessary to carefully define and select which part of the results has to be shown to the experts. Another limitation is that while we allow the links between objects to be more or less uncertain, we assume that the scores $S_{i,j}$ are reliable estimators of this uncertainty, as otherwise the discount factor ε^* to get a consistent matrix may be overly high. Solutions to solve this issue could be to allow a limited number of violations of the equivalence relation properties, possibly going towards a "soft" version of our methods.

Acknowledgements

This work was carried out in the framework of 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). The authors wish to thank Jean-Benoist Leger for early discussions about the algorithmic issues solved in this work.

References

Bartlett, P.L., Wegkamp, M.H., 2008. Classification with a reject option using a hinge loss. Journal of Machine Learning Research 9, 1823–1840.

Ben-Dor, A., Shamir, R., Yakhini, Z., 1999. Clustering gene expression patterns. Journal of computational biology 6, 281–297.

Bezdek, J., 1981. Pattern Recognition with Fuzzy Objective Function Algorithms. Plenum Press, New York.

Biernacki, C., Celeux, G., Govaert, G., 2000. Assessing a mixture model for clustering with the integrated completed likelihood. IEEE transactions on pattern analysis and machine intelligence 22, 719–725.

Carrington, P.J., Scott, J., Wasserman, S., 2005. Models and methods in social network analysis. volume 28. Cambridge university press.

Carvalho, F.D., Lechevallier, Y., 2009a. Dynamic clustering of interval-valued data based on adaptive quadratic distances. IEEE Transactions on Systems, Man, and Cybernetics - Part A: Systems and Humans 39, 1295–1306.

Carvalho, F.D., Lechevallier, Y., 2009b. Partitional clustering algorithms for symbolic interval data based on single adaptive distances. Pattern Recognition 42, 1223–1236.

Carvalho, F.D., Souza, R.D., Chavent, M., Lechevallier, Y., 2006. Adaptative Hausdorff distances and dynamic clustering of symbolic interval data. Pattern Recognition Letters 27, 167–179.

Cazes, P., Chouakria, A., Diday, E., Schektman, Y., 1997. Extension de l'analyse en composantes principales à des données de type intervalle. Revue de Statistique Appliquée 14, 5–24.

Cheng, W., Hüllermeier, E., Waegeman, W., Welker, V., 2012. Label ranking with partial abstention based on thresholded probabilistic models, in: Advances in neural information processing systems, pp. 2501–2509.

De Oliveira, J.V., Pedrycz, W., 2007. Advances in fuzzy clustering and its applications. John Wiley & Sons.

Denoeux, T., Masson, M., 2004. EVCLUS: evidential clustering of proximity data. IEEE Transactions on Systems, Man, and Cybernetics, Part B 34, 95–109.

Fagiolo, G., Mastrorillo, M., 2013. International migration network: Topology and modeling. Physical Review E 88, 012812.

Figueiredo, R., Frota, Y., 2014. The maximum balanced subgraph of a signed graph: applications and solution approaches. European Journal of Operational Research 236, 473–487.

Fitch, W.M., Margoliash, E., 1967. Construction of phylogenetic trees. Science 155, 279–284.

He, L., Ren, X., Gao, Q., Zhao, X., Yao, B., Chao, Y., 2017. The connected-component labeling problem: a review of state-of-the-art algorithms. Pattern Recognition 70, 25–43.

Lichman, M., 2013. UCI machine learning repository. URL: http://archive.ics.uci.edu/ml.

Lingras, P., 2001. Unsupervised rough set classification using GAs. Journal of Intelligent Information Systems 16, 215–228.

Long, B., Zhang, Z.M., Yu, P.S., 2007. A probabilistic framework for relational clustering, in: Proceedings of the 13th ACM SIGKDD international conference on Knowledge discovery and data mining, ACM. pp. 470–479.

Masson, M., Denoeux, T., 2008. ECM: An evidential version of the fuzzy c-means algorithm. Pattern Recognition 41, 1384–1397.

Masson, M., Denoeux, T., 2009. RECM: Relational evidential c-means algorithm. Pattern Recognition Letters 30, 1015–1026.

Melendez-Melendez, G., Cruz-Paz, D., Carrasco-Ochoa, J.A., Martínez-Trinidad, J.F., 2019. An improved algorithm for partial clustering. Expert Systems with Applications 121, 282–291.

Rand, W., 1971. Objective criteria for the evaluation of clustering methods. Journal of the American Statistical Association 66, 846–850.

Rendle, S., Schmidt-Thieme, L., 2008. Active learning of equivalence relations by minimizing the expected loss using constraint inference, in: Data Mining, 2008. ICDM'08. Eighth IEEE International Conference on, IEEE. pp. 1001–1006.

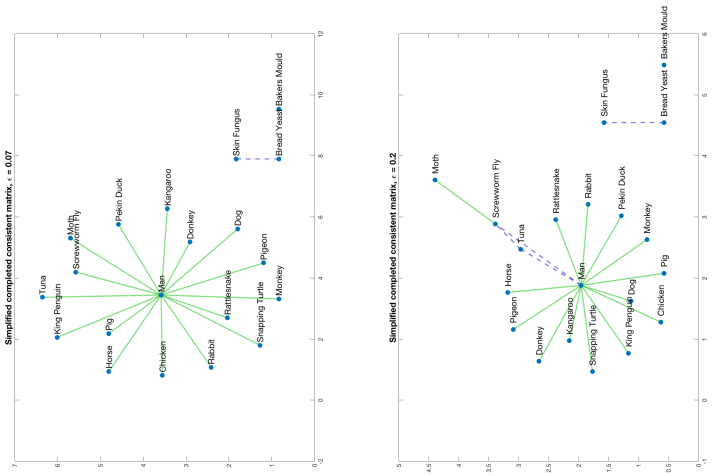
Tellaroli, P., Bazzi, M., Donato, M., Brazzale, A.R., Drăghici, S., 2016. Cross-clustering: a partial clustering algorithm with automatic estimation of the number of clusters. PloS one 11, e0152333.

Ünlü, R., Xanthopoulos, P., 2019. Estimating the number of clusters in a dataset via consensus clustering. Expert Systems with Applications 125, 33–39.

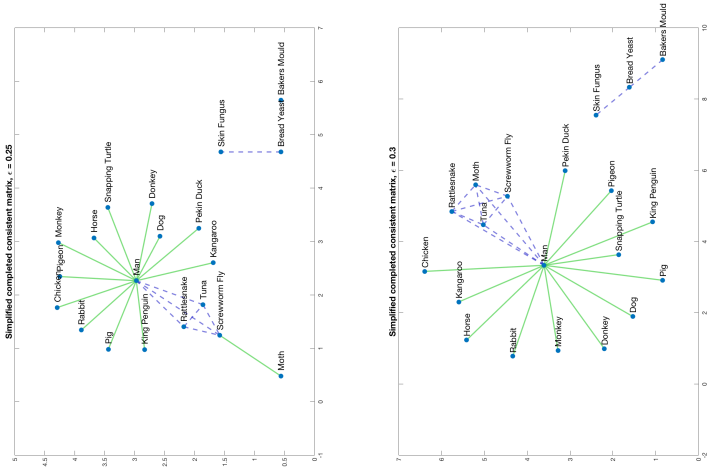
Yang, G., Destercke, S., Masson, M.H., 2017. Cautious classification with nested dichotomies and imprecise probabilities. Soft Computing 21, 7447–7462.

Zhu, S., Xu, L., 2018. Many-objective fuzzy centroids clustering algorithm for categorical data. Expert Systems with Applications 96, 230–248.

Appendix A. Mutation data set



(a) Minimal correction ($\varepsilon = 0.07$) (b) $\varepsilon = 0.20$



(c) $\varepsilon = 0.25$ (d) $\varepsilon = 0.30$

Figure A.11: Partial clusterings of the Mutation data.

Appendix B. Airport data sets

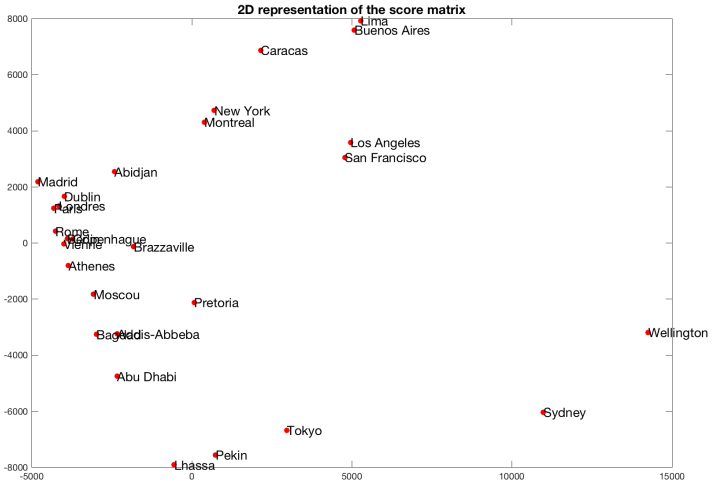
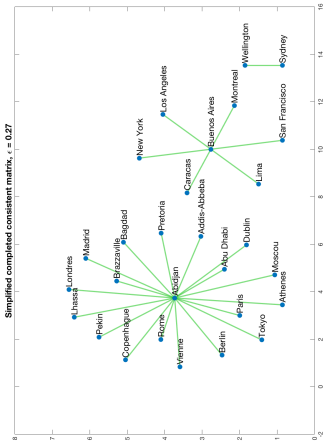
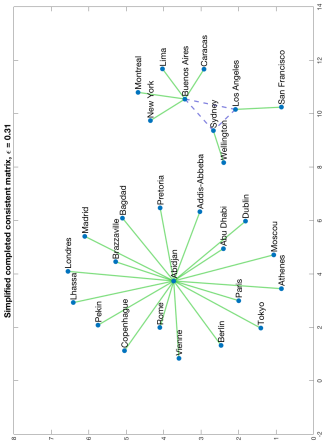


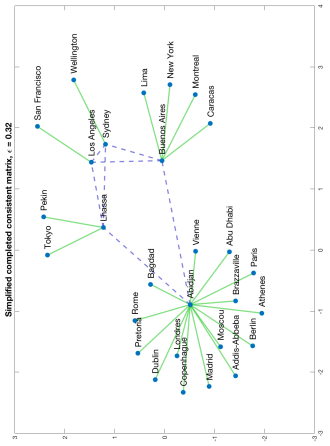
Figure B.12: Representation of the Airports data.



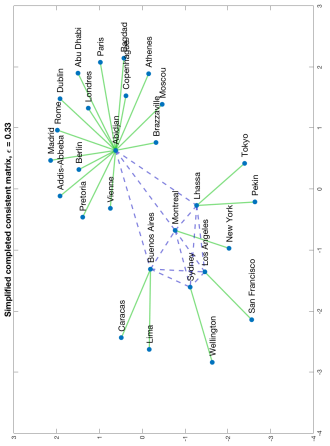
(a) Minimal correction ($\varepsilon = 0.27$)



(b) $\varepsilon = 0.31$



(c) $\varepsilon = 0.32$



(d) $\varepsilon = 0.33$

Figure B.13: Partial clusterings of the Airports data.

Appendix C. Completeness-correctness tradeoff

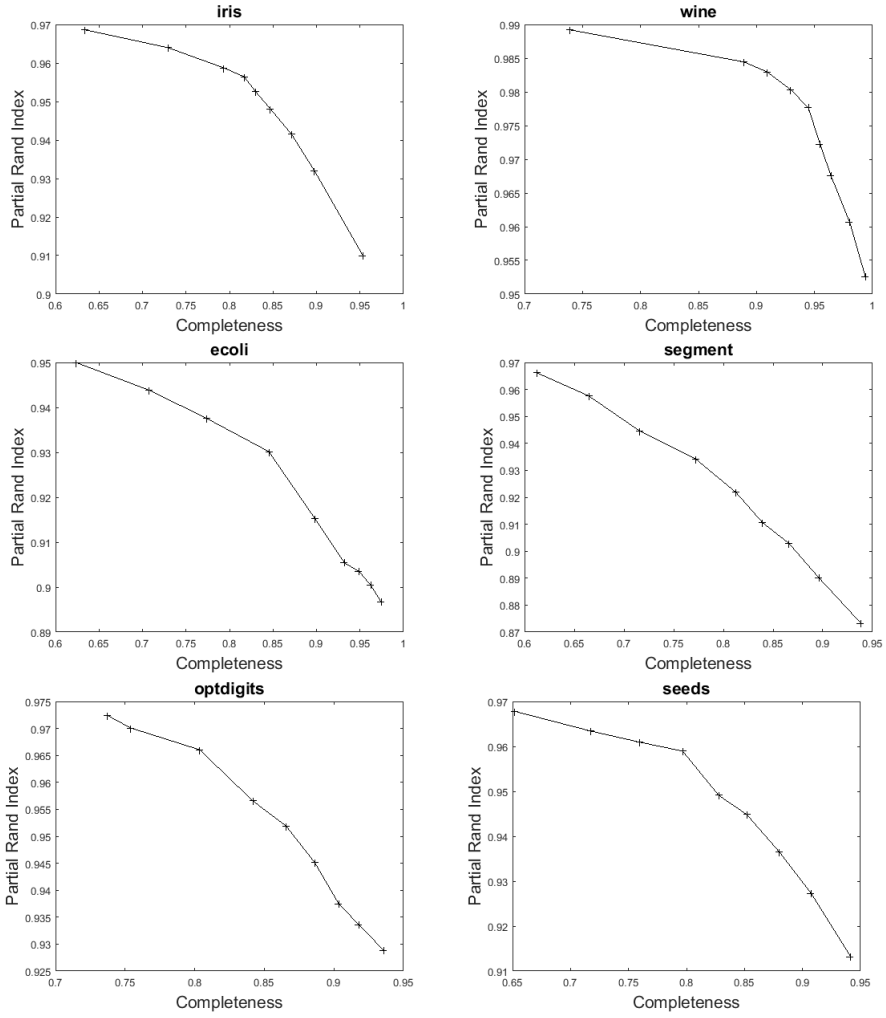


Figure C.14: Clustering results on 6 real datasets