Towards the Re-identification of Individuals in Data Files with Non-common Variables

Vicenç Torra [1]

Abstract. Record linkage is used to establish relationships between records of two different data files. In this work, record linkage is studied for files that correspond to the same set of individuals but that do not share a common set of variables. Under this circumstance, classical techniques can not be applied. We present an approach to this problem based on clustering techniques and knowledge integration ones. In this way, common underlying structures in both files can be detected and re-identification is possible. This approach is based on some basic assumptions that are made explicit in this work.

1 INTRODUCTION

The main task of National Statistical Offices (NSO) is to collect information from individuals and organizations and disseminate this information for researchers, media and general public. The dissemination of the information is sometimes problematic due to disclosure risk. Disclosure risk is defined as the risk of re-identification of particular individuals. This is, some sensitive and confidential data that have been released are afterwards identified with a particular individual and, thus, confidentiality is lost. To avoid re-identification, data is distorted before its release. In this way, disclosure risk decreases. However, data has to maintain [1] the so-called analytical validity, this is, data after being distorted has to reproduce the statistical analysis that can be produced with the original confidential data.

Data distortion is measured with information loss measures. Note that a large distortion implies that analytical validity is lost, and thus implicit information in the initial data is also lost. Moreover, when data distortion increases, disclosure risk decreases. So, as reported in [2] both the information loss and the disclosure risk associated to the released data should be kept small.

Among the existing re-identification methods, we underline record linkage. Record linkage [3] is used to link records in separate files that relate to the same individual or household. These methods [4], [1], [3] are based on the presence of a set of common variables in both files. The main difficulty that face all these methods is that a matching procedure among pairs of records is not always enough to establish the link between the records. As [1] points out, "the normal situation in record linkage is that identifiers in pairs of records that are truly matches disagree by small or large amounts and that different combinations of the non-unique, error-filled identifiers need to be used in correctly matching different pairs of records".

Record linkage, as a method for re-identification of individuals in data files, is a threat for NSO. It can disclosure sensible data when released records are linked with public information. However, besides of being a threat, it is also a solution for some of the problems that face companies and organizations. In particular, organizations often have non-homogeneous distributed data bases. For example, information on customers and suppliers is often distributed over several departments; data is stored in different platforms; data is not standardized (e.g. names and addresses have been written, and sometimes shortened, using different conventions). Under these circumstances, the result of a query can be incorrect (or inconsistent with previous results) as data bases are not complete and they do not satisfy entity integrity. In this environment, record linkage techniques are applied to palliate the inconsistency of data bases. They link records that belong to different files (or databases) corresponding to a single entity.

Recent developments in this area are described in [1], [3], [5], [6]. The latter work compares some of the existing methods. Usually these kind of systems (e.g. Integrity [7]) use Statistical and Artificial Intelligence methods to determine the matching between records and to extract a unique identifier (or a set of variables acting as an identifier). Although most of the methods consider the case when files share a set of common variables, other situations are of interest. In particular, we are interested in the case of non common variables (but still sharing individuals). This is of relevance when considering data files with similar information (e. g. economical variables) from consecutive time periods (e.g. two different years) concerning to almost the same individuals (e. g. the companies of a certain region). In this case, although the variables are not the same, re-identification and record linkage is possible as we show in this work. Although, at present, no much effort has been devoted to this research subject, it is a subject of interest to both areas of statistical offices and data mining. The latter because the methods can be applied to bring into light relationships between individuals that otherwise would remain implicit in the files.

Re-identification can be analyzed from the perspective of knowledge elicitation from groups of experts as in [8] or [9]. In these two works, a common conceptual structure is built from the information supplied by the group of experts and this common structure should correspond to the ones obtained from the experts. In our approach to re-identification, we assume that this common structure has to exist in order to establish the link between the individuals in both files. Besides of this relation between both areas, also the initial information in both problems is similar: records corresponding to objects and evaluated according to a set of variables.

Following the terminology in [9] for knowledge elicitation from groups, four cases are of consideration according to the coincidence or non-coincidence of variables (distinctions in [9] work) and terminology (the domain of the variables, i. e. the terms used to evaluate the individuals): Consensus, correspondence, conflict and contrast. Classical record linkage is placed in the case of consensus (equal variables and terminology) or correspondence (equal vari-

[1] Institut d'Investigació en Intel.ligència Artificial - CSIC, Campus UAB, 08193 Bellaterra (Catalunya, Spain), e-mail: vtorra@iiia.csic.es

ables but different terminology) although the latter is only allowed for small differences on terminology (small inconsistencies among names, missing values and the like). However, based on this classification other types of record linkage are advisable: Correspondence when the degree of non-coincidence on the terminology is not limited to small variations of names (e. g. completely different terms, due, for example, to the use of different granularities), contrast (different variables and different terminology) and conflict (different variables but with equal terminology).

We study here the case of contrast. This is, when there is no coincidence neither on the variables nor on the terminology. However, to do so, we assume that a common underlying structure exists in both files. This is a basic assumption in our approach. Reconsidering the case given above (economical variables of companies in two different years), this means that, for example, when companies are similar in relation to some economic indices for the first year, then they are also similar in relation to the ones for the second year.

In this work, we review in Section 2 some results needed in the rest of the paper. In Section 3 we present the basic assumptions we require for re-identification and a method for re-identification based on clustering techniques. Section 4 show the results of our approach for an example. The paper finishes, Section 5, with some conclusions and future work.

2 PRELIMINARIES

In this section we review some of the theoretical results that belong to the area of aggregation of equivalence relations. Equivalence relations are of interest here as a way to express relationship between objects (they are equivalent to partition of objects). Moreover, clustering techniques permits to obtain partitions from sets of objects.

Definition 1 *A binary relation R on a set A is an equivalence relation if and only if, for all a, b, c in A the following conditions hold: (i) reflexivity: $R(a, a)$; (ii) simmetry: $R(a, b)$ implies $R(b, a)$; (iii) transitivity: $R(a, b)$ and $R(b, c)$ implies $R(a, c)$.*

Definition 2 *An aggregation function $\mathbb{C}$ over equivalence relations on A is a function that given n equivalence relations $R_1, ..., R_n$ defines a new equivalence relation on A. We denote the aggregated relation by: $\mathbb{C}(R_1, ..., R_n)$.*

Although it is usual to add some restrictive conditions (e. g. unanimity: when $R_1 = R_2 = ... = R_n = R$ then $\mathbb{C}(R_1, ..., R_n) = R$) to define what an aggregation function is (see [10] for examples in the ordinal case), we do not follow this approach here. All conditions are established in the next Theorem.

Definition 3 *Let $R_1, ..., R_n$ be relations on A, then an aggregation function $\mathbb{C}$ is a conjunctive aggregation function if there exists a nonempty subset N of $\{1, .., n\}$ such that*
$R(a, b)$ if and only if $R_i(a, b)$ for every $i \in N$
being $R = \mathbb{C}(R_1, ..., R_n)$.

Definition 4 *An aggregation function is said to be consistent if it satisfies the following two conditions:*

1. *For all $a, b \in A$ and all n-tuples $(R_1, ..., R_n)$ and $(R'_1, ..., R'_n)$ of equivalence relations on A,*
   if $R_i(a, b)$ if and only if $R'_i(a, b)$ for $i = 1, ..., n$,
   then $R(a, b)$ if and only if $R'(a, b)$
   being $R = \mathbb{C}(R1, ..., Rn)$ and $R' = \mathbb{C}(R'1, ..., R'n)$
2. *For all $a, b \in A$ and all $(R_1, ..., R_n)$,*
   if $R_i(a, b)$ for $i = 1, ..., n$ then $R(a, b)$, and
   if not $R_i(a, b)$ for $i = 1, ..., n$ then not $R(a, b)$.
   being, as before, $R = \mathbb{C}(R_1, ..., R_n)$.

The first condition in the above definition is the so-called independence condition. It means that the aggregated relation value between a and b does only depend on the relations between these two elements. This is equivalent to say that the aggregation can be computed over pairs. This is, there exists a function F such that:

$$R(a, b) = F(R_1(a, b), ..., R_n(a, b))$$

The second condition is unanimity over pairs. This is, when all equivalence relations relate a and b the aggregated one also relate them. On the other hand, when all equivalence relations say that a and b are not related, the aggregated relation does not relate them.

Theorem 1 *[11]. When A has at least three elements, the set of consistent aggregators equals the set of conjunctive aggregation functions.*

This theorem implies that selecting a subset N of $\{1, ..., n\}$ the aggregation function is determined. Moreover, as [11] reports, it says that all i not in N are irrelevant to the aggregate classification while every $i \in N$ is essential and equally weighted.

3 ON THE RE-IDENTIFICATION PROCESS

As told in the first Section, to allow for re-identification when no common variables exist, some basic assumptions have to be made. We summarize them as follows:

Hypothesis 1 *A large set of common individuals is shared by both files.*

Hypothesis 2 *Data in both files contain, implicitly, similar structural information.*

Structural information (of data files) stands in our case to any organization of the data that allows us to make explicit the relationship between individuals. This structural information is obtained from the data files through manipulation of the data in the file (e.g. using clustering techniques or any other data analysis or data mining technique). The comparison of the structural information implicit in both files is what allows the system to link two records that correspond to the same individual.

In our approach the structural information is assumed to be represented by means of partitions. Partitions obtained from data by means of clustering techniques make implicit the relation between individuals according to the variables that describe them. Common partitions in both files correspond to the common structural information. We use partitions instead of other more sophisticated structures (like dendrograms) also obtainable from cluster methods because the former are less sensitive to changes in the data. Therefore, they can led to better results. Results on consensus of classifications [12] support this approach.

Hypothesis 3 *Structural information can be expressed by means of partitions.*

Although the main interest of our research is to re-identify individuals, the approach described below is not directly focused on the re-identification of particular individuals. Instead, we try to re-identify groups of them. We consider a two stages approach. On a first stage, groups of individual are re-identified and after that, in a second stage, individuals are identified with a detailed analysis of the data. Due to this, we use the terminology of re-identification at the group level and at the individual level. This work is limited to group level re-identification.

3.1 Identification at the group level

In the group level, general structural information is identified in both files by means of clustering techniques. Moreover, as different clustering techniques identify different relationships between the individuals, several ones are applied (different ones or the same ones but changing parameters) to both files. In this way, we obtain for each file and each technique a partition of the individuals. This initial process is formalized below considering that data files are named A and B, and as usual, files are defined by a set of records that assign values to variables. We assume that the file with known individuals is the file B.

Table 1. Partitions obtained from file A with clustering techniques $\{CP = CP_1, .., CP_t\}$

$FileA$	CP_1	CP_2		CP_t
π_1^A	$c_{1,1}^A$	$c_{2,1}^A$	...	$c_{t,1}^A$
...	...	...		...
$\pi_{p(A)}^A$	$c_{1,p(A)}^A$	$c_{2,p(A)}^A$	...	$c_{t,p(A)}^A$

Table 2. Partitions obtained from file B with clustering techniques $CP = \{CP_1, .., CP_t\}$

$FileB$	CP_1	CP_2		CP_t
π_1^B	$c_{1,1}^B$	$c_{2,1}^B$	...	$c_{t,1}^B$
...	...	...		...
$\pi_{p(B)}^B$	$c_{1,p(B)}^B$	$c_{2,p(B)}^B$	...	$c_{t,p(B)}^B$

Let $V_A = \{A_1, ..., A_{n(A)}\}$ and $V_B = \{A_1, ..., A_{n(B)}\}$ be the set of variables of data files A and B, respectively. The number of variables ($n(A)$ and $n(B)$), and the variables themselves are not needed to be the same in both files. Let $O_A = \{O_1^A, ..., O_{m(A)}^A\}$ and $O_B = \{O_1^B, ..., O_{m(B)}^B\}$ be the objects in both files. As with the number of variables, we do not require that the number of objects ($m(A)$ and $m(B)$) are the same. Although according to Hypothesis 1 a large number of individuals is shared in both files, it is neither known which are the common individuals nor are they identified. Therefore, at this point, different names apply to them.

To each file we apply a series of clustering process. This is, we consider a set of t different clustering methods (or the same clustering method with different parameters) and we apply each of them to each file. Each method leads to a partition of the domain (a set of classes). Let $CP = \{CP_1, ..., CP_t\}$ be the set of clustering techniques considered, let $C_{i,A} = \{C_{i,A,1}, ..., C_{i,A,nc(i)}\}$ and $C_{i,B} = \{C_{i,B,1}, ..., C_{i,B,nc(i)}\}$ be the clusters obtained when the clustering technique CP_i is applied to files A and B, respectively. Here, $C_{i,A,j}$ and $C_{i,B,j}$ correspond to the j-th cluster obtained by the clustering technique CP_i when applied to data files A and B. For the sake of simplicity, $C_{i,A,j}$ and $C_{i,B,j}$ are subsets of the objects in the file A and B. This is, $C_{i,A,j} \subseteq O_A$ and $C_{i,B,j} \subseteq O_B$. Note that we have assumed that the number of clusters of a clustering technique CP_i is the same when applied to both data files $nc(i)$. This is so to be able to identify the clusters corresponding to one file with the ones in the other file sharing the same objects. Moreover, for several clustering algorithms (e.g. Fuzzy c-means and the ones used in this work) the number of clusters to be obtained is one of the parameters of the system and, thus, this is not a restriction of the methods. $CP_i(O)$ denotes the cluster to which object O belongs (as the object belongs to a single file, it is completely determined the domain of $CP_i(O)$).

Once all partitions have been obtained for each CP_i, we build the structural information for each file. This information is a partition (according to hypothesis 3) and it should synthesize the common information extracted by all CP_i. Due to this, we combine all partitions $C_{i,A}$ and $C_{i,B}$ by means of an aggregation function. In particular, and according to Theorem 1 in Section 1 (that states that the only consistent aggregation functions are conjunctive ones), we use the conjunction of all partitions. Thus, the structural information of a file is defined as the intersection of all partitions obtained by the clustering methods in CP. Let $\Pi(A, CP)$ and $\Pi(B, CP)$ be the structural information of files A and B. They are defined as follows:

$$\Pi(A, CP) = \{\cap_{i,j} C_{i,A,j}\}$$

$$\Pi(B, CP) = \{\cap_{i,j} C_{i,B,j}\}$$

We will denote the elements in $\Pi(A, CP)$ and $\Pi(B, CP)$ as follows:

$\Pi(A, CP) = \{\pi_1^A, ..., \pi_{p(A)}^A\}$, $\Pi(B, CP) = \{\pi_1^B, ..., \pi_{p(B)}^B\}$.

From the above definition the next proposition follows:

Proposition 1 *If $\Pi(A, CP) = \{\pi_1^A, ..., \pi_{p(A)}^A\}$ and $\Pi(B, CP) = \{\pi_1^B, ..., \pi_{p(B)}^B\}$ as defined above, then they are partitions of O_A and O_B.*

Proposition 2 *The following two conditions hold for $\Pi(A, CP)$: (i) all objects in the same partition element π are clustered together in all partitions $C_{i,A}$; (ii) two individuals in two different clusters π_i and π_j are, at least for a clustering method, clustered in different partitions. The same applies for $\Pi(B, CP)$.*

For each π_i^A and π_i^B, we denote by $CP_i(\pi_j^A) = c_{i,j}^A$ the class to which all the objects in π_j^A belong. Tables 1 and 2 put together all this notation. Due to the definition of Π, all objects in π_j^A belong to the same classes for all the methods. The same conditions apply to the data file B. This is,

for all $o_A \in \pi_j^A$ and for all $i \in \{1, ..., t\}$, $CP_i(o_A) = CP_i(\pi_j^A)$

for all $o_B \in \pi_j^B$ and for all $i \in \{1, ..., t\}$, $CP_i(o_B) = CP_i(\pi_j^B)$

To put both files into correspondence, we need to associate for each cluster in one data file a cluster in the other one. However, as association has to be made cluster to cluster, a mapping for each clustering technique is needed. Therefore, we consider for each $C_{i,A}$ a function f_i that assigns a cluster in $C_{i,B}$ to each of the clusters in $C_{i,A}$. I.e., $f_i : C_{i,A} \rightarrow C_{i,B}$ for all $i \in \{1, ..., t\}$. These functions have to be defined so that when applied to Table 1 return a table that is as similar as possible to Table 2. We define similarity between tables on the row basis: a row in the first table (e.g. the one of the π_i^A) should be similar to one of the rows in the second table. This is, it should exist a π_k^B in $\Pi(B, CP)$ similar to $f(\pi_i^A) = (f_1(c_{i,1}^A), ..., f_t(c_{i,t}^A))$ for all i. Other definitions of

similarity could apply; however, as the goal is to re-identify clusters and, in particular, elements in $\Pi(A, CP)$ with other elements in $\Pi(B, CP)$ from our point of view $\pi \in \Pi$ seem to be the basic units.

Based on this assumption, we have used the following similarity function $S : [C_{i,B} \times ... \times C_{i,B}]^2 \rightarrow \mathbb{R}$.

$S(X = (x_1, x_2, ..., x_t), Y = (y_1, y_2, ..., y_t)) = \Sigma d_i(x_i, y_i)$, where $d_i(x_i, y_i) = 1$ if $x_i = y_i$, and 0 otherwise.

To finish the formalization of the re-identification process we need the group level re-identification function (i. e., the one that relates the i-th partition of $\Pi(A, CP)$ with the k-th partition of $\Pi(B, CP)$). We call this function m and takes the form: $m : \{1, ..., p(A)\} \rightarrow \{1, ..., p(B)\}$.

Putting all this together, the group level re-identification problem can be formulated in the following way:

The group level re-identification problem: Find functions $f = (f_1, ..., f_t)$ and m such that the $\Sigma_{i=1,t} S(f(\pi_i^A), \pi_{m(i)}^B)$ is maximized.

In relation to this formulation, the following remarks are of interest:

1. The assumption that the file with known records is B is present on the re-identification function m that is from clusters in A into the ones in B and on the similarity function that is computed comparing partitions in B.
2. Restrictions over f_i can apply. In particular, we can consider that: $(f_i(C_{i,A,1}), ..., f_i(C_{i,A,nc(i)}))$ is a permutation of the vector $(C_{i,B,1}, ..., C_{i,B,nc(i)})$.
3. $p(A)$ can be different to $p(B)$. Therefore, m is not always a one-to-one function.
4. When the number of clustering methods increases (larger sets CP), the number of elements in $\Pi(A, CP)$ and $\Pi(B, CP)$ tend to be the number of elements in the files. Therefore, re-identification tend to be at the individual level.
5. The formulation of the problem allows the user to consider also common variables, specially, with different terminology (i.e., correspondence).

4 RESULTS

In the following we show the feasibility of the approach, analyzing a small artificial problem. The example, that uses publicly available data, details all the steps described in the previous section.

Example 1 *To test the methodology with a well-known and public data file, we have considered the ionosphere data base in the UCL repository [13]. This example consists of a set of 351 examples (positive and negative examples) each defined in terms of 34 numerical variables.*

To use this data for re-identification, two alternatives were possible: re-identification of the examples and re-identification of the variables. In the first alternative, the original file would be split so that all examples were present in both files, but only half of the variables would be present in each file. In the second alternative, the original file would be split so that all variables were present in both files but, instead, only half of the examples. We have followed the latter approach because it is not sure that half of the variables have enough information about the examples to allow for re-identification. Instead, two randomly chosen subsets of about 175 examples should give enough information about the structure of the variables. In fact, subsets of these examples are usually used in machine learning [14]. They assume that subsets have still enough information on the variables. In other words, this approach was used because we assumed more redundancy in the examples than in the variables.

To apply the method described above, we have considered an initial normalization step. It consisted on the normalization of the domain of the variables and it was applied before the file was partitioned (usual normalization in the $[0, 1]$ interval was applied: $x' = (x - min)/(max - min)$). Then, the set of examples was randomly divided into two sets. One set resulted with 170 and the other with 181 examples.

Table 3. Partitions $\Pi(A, CP)$, two group level identification m_1 and m_2, and the corresponding partitions $CP_i(\pi_j^A)$. In the last six columns, aa and cc correspond to the classification criteria and stand for Arithmetic average and Centroid clustering; m, d and t refer to similarity functions based, respectively, on Manhattan distance, Differences and Taxonomic distance. In this table $c_{r,s}$ stand for $c_{r,A,s}$.

	m_1	m_2	aa,m	aa,d	aa,t	cc,m	cc,d	cc,t
O_1^A	1	1	$c_{1,4}$	$c_{2,4}$	$c_{3,1}$	$c_{4,1}$	$c_{5,3}$	$c_{6,3}$
O_2^A	2	2	$c_{1,1}$	$c_{2,2}$	$c_{3,4}$	$c_{4,4}$	$c_{5,3}$	$c_{6,5}$
O_4^A	3	3	$c_{1,2}$	$c_{2,2}$	$c_{3,}$	$c_{4,2}$	$c_{5,6}$	$c_{6,2}$
O_5^A	4	4	$c_{1,4}$	$c_{2,2}$	$c_{3,1}$	$c_{4,3}$	$c_{5,6}$	$c_{6,1}$
$O_6^A O_8^A$	6	6	$c_{1,2}$	$c_{2,2}$	$c_{3,2}$	$c_{4,2}$	$c_{5,4}$	$c_{6,2}$
$O_7^A O_9^A$	5	5	$c_{1,4}$	$c_{2,2}$	$c_{3,1}$	$c_{4,3}$	$c_{5,4}$	$c_{6,1}$
O_{10}^A	10	10	$c_{1,2}$	$c_{2,1}$	$c_{3,2}$	$c_{4,2}$	$c_{5,3}$	$c_{6,2}$
$O_3^A O_{11}^A$	7	7	$c_{1,4}$	$c_{2,2}$	$c_{3,1}$	$c_{4,3}$	$c_{5,3}$	$c_{6,1}$
O_{14}^A	10	10	$c_{1,2}$	$c_{2,1}$	$c_{3,2}$	$c_{4,2}$	$c_{5,5}$	$c_{6,2}$
O_{15}^A	14	14	$c_{1,4}$	$c_{2,1}$	$c_{3,1}$	$c_{4,3}$	$c_{5,5}$	$c_{6,1}$
$O_{12}^A O_{16}^A$	11	11	$c_{1,2}$	$c_{2,1}$	$c_{3,2}$	$c_{4,2}$	$c_{5,4}$	$c_{6,2}$
O_{18}^A	16	13	$c_{1,3}$	$c_{2,1}$	$c_{3,2}$	$c_{4,2}$	$c_{5,1}$	$c_{6,2}$
O_{19}^A	17	21	$c_{1,4}$	$c_{2,1}$	$c_{3,1}$	$c_{4,3}$	$c_{5,1}$	$c_{6,1}$
O_{20}^A	15	13	$c_{1,3}$	$c_{2,1}$	$c_{3,2}$	$c_{4,2}$	$c_{5,2}$	$c_{6,2}$
O_{21}^A	21	21	$c_{1,4}$	$c_{2,1}$	$c_{3,1}$	$c_{4,3}$	$c_{5,3}$	$c_{6,1}$
$O_{22}^A O_{24}^A$	18	18	$c_{1,3}$	$c_{2,1}$	$c_{3,2}$	$c_{4,2}$	$c_{5,4}$	$c_{6,2}$
$O_{25}^A O_{13}^A O_{17}^A$	20	21	$c_{1,4}$	$c_{2,1}$	$c_{3,1}$	$c_{4,3}$	$c_{5,2}$	$c_{6,1}$
$O_{23}^A O_{27}^A$	19	17	$c_{1,4}$	$c_{2,1}$	$c_{3,1}$	$c_{4,3}$	$c_{5,4}$	$c_{6,1}$
O_{28}^A	22	22	$c_{1,3}$	$c_{2,3}$	$c_{3,2}$	$c_{4,2}$	$c_{5,6}$	$c_{6,2}$
$O_{29}^A O_{31}^A$	19	19	$c_{1,4}$	$c_{2,3}$	$c_{3,1}$	$c_{4,3}$	$c_{5,4}$	$c_{6,1}$
$O_{26}^A O_{30}^A$								
O_{32}^A	22	22	$c_{1,3}$	$c_{2,3}$	$c_{3,2}$	$c_{4,2}$	$c_{5,4}$	$c_{6,2}$
O_{33}^A	21	21	$c_{1,4}$	$c_{2,3}$	$c_{3,1}$	$c_{4,3}$	$c_{5,6}$	$c_{6,1}$
O_{34}^A	22	22	$c_{1,3}$	$c_{2,3}$	$c_{3,2}$	$c_{4,2}$	$c_{5,7}$	$c_{6,2}$
O_{35}^A	23	23	$c_{1,4}$	$c_{2,3}$	$c_{3,3}$	$c_{4,1}$	$c_{5,4}$	$c_{6,4}$

The next step was to obtain the partitions as in Tables 1 and 2. To do so, six classification techniques were applied to both files. Each technique led to a dendrogram, and for each dendrogram a partition was obtained. Dendrograms were obtained using SAHN [15] methods (i.e., sequential, agglomerative, hierarchic, non-overlapping methods). Different selection of similarity functions (functions to compute similarities between objects/classes) and of classification criteria (how to compute, from already known similarities, the similarity between a new class and the previous existing ones) were applied. Three similarity functions were used (based on Manhattan distance, Differences and Taxonomic distance) combined with two classification criteria (Arithmetic average, centroid clustering). The definition of these functions and their properties are detailed in [15]. Once the partitions were build for the six methods, the partitions $\Pi(A, CP)$ and $\Pi(B, CP)$ were built. These latter partitions together with the partitions obtained by the six classification methods are given in Tables 3 and 4.

With all this information, we have solved the maximization problem formalized above. Two solutions are given. One restricting f_i to be a one-to-one function and another one allowing f_5 to be such

that $f_5(x) = f_5(y)$ when $x \neq y$. The functions f_i of the solution are given in Table 5 while the group level re-identification functions are given in Table 3. In this latter table, m_1 correspond to the function with f_5 being a one-to-one function while m_2 corresponds to the other case.

Once the method has been applied, we have compared its performance with respect to correct re-identifications. To do so, we have computed the number of objects that belong to the class pointed out by the re-identification function (m_1 and m_2, respectively). This has been normalized by the number of elements so that the function has a maximum value of 1. This is:

$$|\{O_i^A | O_i^A \in \pi^B_{m(i)}\}|/|O_A|$$

where m is the re-identification function. It can be observed that in both cases, 20 out of 35 objects have been correctly re-identified. Thus, 57% of the objects have been correctly re-identified. In both case the similarity function S between the two partitions is 137. Correctly re-identified objects are not the same for both cases.

Table 4. Partitions $\Pi(B, CP)$ and the corresponding partitions $CP_i(\pi_j^B)$. Meanings of aa, cc, m, d and t as in Table 3. In this table $c_{r,s}$ stand for $c_{r,B,s}$.

		aa,m	aa,d	aa,t	cc,m	cc,d	cc,t
1	O_1^B	$c1,1$	$c2,4$	$c3,1$	$c4,3$	$c5,5$	$c6,3$
2	O_2^B	$c1,4$	$c2,2$	$c3,4$	$c4,4$	$c5,5$	$c6,5$
3	$O_4^B O_6^B$	$c1,2$	$c2,2$	$c3,3$	$c4,1$	$c5,2$	$c6,2$
4	$O_5^B O_7^B$	$c1,1$	$c2,2$	$c3,1$	$c4,2$	$c5,2$	$c6,1$
5	O_9^B	$c1,1$	$c2,2$	$c3,1$	$c4,2$	$c5,1$	$c6,1$
6	$O_8^B O_{10}^B$	$c1,2$	$c2,2$	$c3,3$	$c4,1$	$c5,1$	$c6,2$
7	$O_3^B O_{11}^B$	$c1,1$	$c2,2$	$c3,1$	$c4,2$	$c5,5$	$c6,1$
8	O_{12}^B	$c1,2$	$c2,2$	$c3,3$	$c4,1$	$c5,7$	$c6,2$
9	O_{13}^B	$c1,1$	$c2,2$	$c3,1$	$c4,2$	$c5,3$	$c6,1$
10	O_{14}^B	$c1,2$	$c2,1$	$c3,3$	$c4,1$	$c5,3$	$c6,2$
11	O_{16}^B	$c1,2$	$c2,1$	$c3,3$	$c4,1$	$c5,1$	$c6,2$
12	O_{20}^B	$c1,3$	$c2,1$	$c3,3$	$c4,1$	$c5,3$	$c6,2$
13	$O_{18}^B O_{22}^B$	$c1,3$	$c2,1$	$c3,3$	$c4,1$	$c5,2$	$c6,2$
14	$O_{15}^B O_{19}^B$						
	O_{23}^B	$c1,1$	$c2,1$	$c3,1$	$c4,2$	$c5,3$	$c6,1$
15	O_{24}^B	$c1,3$	$c2,1$	$c3,3$	$c4,1$	$c5,4$	$c6,2$
16	O_{26}^B	$c1,3$	$c2,1$	$c3,3$	$c4,1$	$c5,6$	$c6,2$
17	O_{27}^B	$c1,1$	$c2,1$	$c3,1$	$c4,2$	$c5,6$	$c6,1$
18	O_{28}^B	$c1,3$	$c2,1$	$c3,3$	$c4,1$	$c5,1$	$c6,2$
19	$O_{17}^B O_{29}^B$	$c1,1$	$c2,1$	$c3,1$	$c4,2$	$c5,1$	$c6,1$
20	$O_{25}^B O_{31}^B$	$c1,1$	$c2,1$	$c3,1$	$c4,2$	$c5,4$	$c6,1$
21	$O_{21}^B O_{33}^B$	$c1,1$	$c2,1$	$c3,1$	$c4,2$	$c5,2$	$c6,1$
22	$O_{30}^B O_{32}^B$						
	O_{34}^B	$c1,3$	$c2,3$	$c3,3$	$c4,1$	$c5,2$	$c6,2$
23	O_{35}^B	$c1,1$	$c2,3$	$c3,2$	$c4,3$	$c5,2$	$c6,4$

Table 5. Functions f_i to maximize the similarity between partitions in Tables 3 and 4. In this table $f_i(c_k)$ stands for $f_i(c_{k,A})$ and $c_{r,s}$ stands for $c_{r,B,s}$.

$f_1(c_1)$	$f_2(c_2)$	$f_3(c_3)$	$f_4(c_4)$	$f_5(c_5)$	$f'_5(c_5)$	$f_6(c_6)$
$c_{1,4}$	$c_{2,1}$	$c_{3,1}$	$c_{4,3}$	$c_{5,6}$	$c_{5,2}$	$c_{6,1}$
$c_{1,2}$	$c_{2,2}$	$c_{3,3}$	$c_{4,1}$	$c_{5,4}$	$c_{5,2}$	$c_{6,2}$
$c_{1,3}$	$c_{2,3}$	$c_{3,2}$	$c_{4,2}$	$c_{5,5}$	$c_{5,5}$	$c_{6,3}$
$c_{1,1}$	$c_{2,4}$	$c_{3,4}$	$c_{4,4}$	$c_{5,1}$	$c_{5,1}$	$c_{6,4}$
				$c_{5,3}$	$c_{5,3}$	$c_{6,5}$
				$c_{5,2}$	$c_{5,2}$	
				$c_{5,7}$	$c_{5,2}$	

5 CONCLUSIONS AND FUTURE WORK

The results given in this work are a first attempt to deal with the problem of re-identification of individuals when non-common variables are shared in two information sources. The results obtained show that it is feasible to deal with this problem, although more extensive research is needed in this direction. In particular, it is needed to analyze which of the clustering techniques are more suitable for re-identification. This need is already suggested by the example described in Section 4 where a detailed analysis of one of the clustering methods (i.e. Centroid clustering with similarity based on differences) show that there is no correlation between partitions for files A and B. Besides of that, further research is needed to identify appropriate algorithms for group level identification and to increase the effectiveness of the system: to obtain and assure the optimal for a particular problem and to compute the optimal with as less partitions as possible.

The approach introduced here has been applied to quantitative data, however the same approach can be applied to data files with other types of data (as ordinal - qualitative) when clustering techniques exists that deals with this data.

ACKNOWLEDGEMENTS

The author is indebted to Josep Domingo and to the referees for helpful comments and suggestions.

REFERENCES

[1] W.E. Winkler, Matching and Record Linkage, in B. G. Cox (ed.), *Business Survey Methods*, J. Wiley, 355–384, (1995).

[2] J. Domingo, V. Torra, On the Connections Between Statistical Disclosure Control for Microdata and Some Artificial Intelligence Tools, (submitted).

[3] J. F. Robinson-Cox, A record-linkage approach to imputation of missing data: analyzing tag retention in a tag-recapture experiment, J. of Agricultural, Bilogical, and Environmental Statistics, **3**, 48–61, (1998).

[4] H. B. Newcombe, J. M. Kennedy, S. J. Axford, A. P. James, Automatic Linkage of Vital Records, *Science*, **130**, 954–959, (1959).

[5] D. L. Rosman, *The linkage of hospital and police information on road crash casualties: an investigation of alternative methods*, Report N. RIIP-7, 1995.

[6] W. E. Winkler, Advanced methods for record linkage, American Statistical Association, *Proceedings of the Section on Survey Research Methods*, 467-472, (1995).

[7] http://www.integrity.com

[8] V. Torra, U. Cortes, Towards an automatic consensus generator tool: EGAC, *IEEE Transactions on Systems, Man and Cybernetics*, **25**, 888–894, (1995).

[9] B. R. Gaines, M. L. G. Shaw, Knowledge Acquisition Tools Based on Personal Construct Psychology, *The Knowledge Engineering Rev.*, **8**, 49–85, (1993).

[10] L. Godo, V. Torra, On aggregation operators for ordinal qualitative information, *IEEE T. Fuzzy Systems*,(in press).

[11] P. C. Fishburn, A. Rubistein, Aggregation of equivalence relations, *J. of classification*, **3**, 61–65, (1986).

[12] D. A. Neumann, V. T. Norton (Jr), Clustering and isolation in the consensus problem for partitions, *Journal of classification*, **3**, 281–297, (1986).

[13] P. M. Murphy, D. W. Aha, UCI Repository machine learning databases, http://www.ics.uci.edu/ mlearn/MLRepository.html, Irvine, CA: University of California, Department of Information and Computer Science, 1994.

[14] V. G. Sigillito, S. P. Wing, L. V. Hutton, K. B. Baker, Classification of radar returns from the ionosphere using neural networks. *Johns Hopkins APL Technical Digest*, **10**, 262–266, (1989).

[15] B. Everitt, *Cluster analysis*, Heinemann Educational Books Ltd, 1977.