Evaluation of Information Loss for Privacy Preserving Data Mining through comparison of Fuzzy Partitions

Isaac Cano, Susana Ladra, Vicenç Torra, *Member, IEEE*

***Abstract*— In this paper, we focus on the problem of preserving the data confidentiality when sharing the data for clustering. This problem poses new challenges for novel uses of privacy preserving data mining (PPDM) techniques. Specifically, this paper considers the synthetic data generation as a way to preserve the data privacy.**

One of the state of the art synthetic data generators is the IPSO family of methods. It has been stated that the use of IPSO to generate synthetic data is appropriate when the user plans to apply clustering to the data. Moreover, this paper aims to associate the same property to the FCRM synthetic data generator, and at the same time, to assess the relationship between the information loss produced when generating synthetic data with FCRM and the clustering similarity between the original and synthetic data.

I. Introduction

The information age has enabled many organizations to gather large volumes of data. However, the usefulness of this data is negligible if *meaningful information* or *knowledge* cannot be extracted from it. Data mining, otherwise known as knowledge discovery, attempts to answer this need. In contrast to standard statistical methods, data mining techniques search for interesting information without demanding a priori hypotheses.

A key problem that arises in any collection of data is that of confidentiality. The need for privacy is sometimes due to law (e.g., for medical databases) or can be motivated by business interests. However, there are situations where the sharing of data can lead to mutual gain. Nevertheless, in this case, confidentiality issues should be taken into account and data mining algorithms should be reconsidered from this point of view. That is, privacy should be preserved.

Privacy preserving data mining [1], [6] is a novel research direction in data mining and statistical databases [26] where data mining algorithms are analyzed for the side-effects they incur in data privacy. The main objective in privacy preserving data mining is to develop algorithms for modifying the original data in some way, so that the private data and private knowledge remain private even after the mining process. The main consideration in privacy preserving data mining is twofold. First, sensitive raw data like identifiers, names, addresses and the like should be modified or trimmed out from the original database, in order for the recipient of the data not to be able to compromise another person's privacy. Second, sensitive knowledge which can be mined from a database by using data mining algorithms should also be excluded, because such knowledge can equally well compromise data privacy.

One approach to privacy preserving data mining is based on cryptography while another approach is based on the perturbation of the data. The former approximation first encrypts the original data, and then performs data mining. Finally the results of the computation are decrypted by the data owners. This cryptographic approach ensures the data privacy insomuch as the whole process is done using encrypted data. The latter approach consists in perturbing the original data, e.g. introducing some kind of noise in them. That is, the perturbed data are released for their analysis. The perturbative approach performs in such a way that the more distortion the data suffers, the less data utility. Because of this, the goal is to achieve a good balance between the level of perturbation, and the data utility, so that the analyses with the original data are equivalent to the ones on the distorted data.

In recent years, the perturbative approach has been expanded with a new research trend, the synthetic data generation. In this case, synthetic data, also considered artificial data, is generated by constructing a model from the original data set and using it to randomly generate a new data set constrained by the model. Although it is possible to publish the model, third parties usually prefer to receive the synthetic data. The Information Preserving Statistical Obfuscation (IPSO) [4] is one of the state of the art synthetic data generators.

In this paper we study the behavior of the synthetic data generated by the fuzzy c-regression models (FCRM), using different data sets, with respect to clustering methods. The aim of this study is to evaluate whether the synthetic data generated with FCRM can be used to perform clustering methods on it. In addition, we want to analyze the relationship between the similarity of the clustering structures obtained when doing clustering with the original and synthetic data, and the information loss incurred when releasing the synthetic data instead of the original data set. We expect that the higher the similarity, the lower the information loss.

This paper is structured as follows. In Section II, we give an overview of the FCRM synthetic data generator. In Section III, we introduce the data utility concept. Then, in Section IV and Section V we present the clustering methods analyzed and the clustering similarity measures used, respectively. Finally, the experiments performed are presented in Section VI and Section VII presents our conclusions.

I. Cano and V. Torra are with the IIIA, Artificial Intelligence Research Institute; CSIC, Spanish National Research Council; Campus UAB; 08193 Bellaterra, Catalonia, Spain. E-mail: cano@iiia.csic.es, vtorra@iiia.csic.es

S. Ladra is with the Database Laboratory; University of A Coruña; Campus de Elvia; 15071 A Coruña, Galiza, Spain. E-mail: sladra@udc.es

II. Fuzzy c-regression

Fuzzy c-regression models (FCRM) are a family of objective functions which can be used to fit switching regression models to numerical and continuous mixed data. For a given c (the number of clusters, $1 < c < n$), the fuzzy c-regression algorithm is able to get an estimation for the parameters of c regression models, together with a fuzzy c-partition of the data. Let us consider a set of object data of size n, $S = \{(x_1, y_1), \ldots, (x_n, y_n)\}$, where each feature vector (x_i, y_i) has a dependent observation $y_i \in \mathbb{R}^t$ corresponding to a certain independent observation $x_i \in \mathbb{R}^s$. The main difference between fuzzy c-regression models and the simplest data fitting problems is that the latter assume that a single functional relationship between x and y holds for all the data while the former assume the data to be drawn from c models:

$$y = f_i(x; \beta_i) + \epsilon, \qquad 1 \leq i \leq c \tag{1}$$

each $\beta_i \in \Omega_i \subset \mathbb{R}^{k_i}$, and each ϵ_i is a random vector with mean vector $\mu_i = 0 \in \mathbb{R}^t$ and covariance matrix Σ_i. It must be told that S is unlabeled, so, for a given feature vector (x_i, y_i), it is not known which model from 1 applies. Hathaway and Bezdek published in [14] a feasible solution for this problem. Their approach is based on fuzzy clustering techniques and is able to produce good estimates of $\{\beta_1, \ldots, \beta_c\}$ while labeling with a fuzzy label vector each datum in S. The labeling problem is solved by means of fuzzy clustering assigning constrained label vectors representing the membership of each object (x_i, y_i) to each of the classes c.

The algorithm for building the Fuzzy c-Regression Models (FCRM) is an iterative process and has the following steps:

1) **Step 1.** Given $S = \{(x_1, y_1), \ldots, (x_n, y_n)\}$ a set of object data. Set $m > 1$ (a reasonable choice is $m = 1.5$), specify regression models (1), and choose a measure of error $E = \{E_{ik}\}$ so that $E_{ik}(\beta_i) \geq 0$ for i and k and also satisfying the minimizer property [14]. Pick a termination threshold $\epsilon > 0$ (a choice for ϵ in the range 0.0001 to 0.00001 usually yields good estimates) and an initial partition $U^{(0)} \in M_f$. In our experiments, we used the Fuzzy c-means [2] algorithm to get such initial partition. Then set a threshold for r_{max}, the maximum number of iterations, so that $r = 1, \ldots, r_{max}$ in case FCRM does not converge (in our experiments a value of $r_{max} = 30$ was used).
2) **Step 2.** Update the values for the c model parameters $\beta_i = \beta_i^{(r)}$ and then the measure of error $E_{ik}(\beta_i)$ in $f_i(x_k; \beta_i)$ that globally minimize (over $\Omega_1 \times \Omega_2 \times \cdots \times \Omega_c$) the restricted function:

$$\psi(\beta_1, \ldots, \beta_c) \equiv E_m(U^{(r)}, \beta_1, \ldots, \beta c)$$

The most common example for the measure of error $E_{ik}(\beta_i)$ is the squared vector norm $E_{ik}(\beta_i) = \|f_i(x_k; \beta_i) - y_k\|^2$. In our case this second step can be specified by fixing $\Omega_i = \mathbb{R}^s$, $f_i(x_k; \beta_i) = ((x_k)^T \beta i)$ and $1 \leq i \leq c$, so, the objective function $E_m(U^{(r)}, \beta_1, \ldots, \beta c)$ becomes a fuzzy multi-model extension of the least squares criterion for model fitting:

$$E_{ik}(\beta_i) = (y_k - (x_k)^T \beta i)^2. \tag{2}$$

In addition, the new values for the regression model parameters $\beta_i^{(r)}$, $1 \leq i \leq c$ can be computed using the following explicit formula if the columns of X are linearly independent and $U_{ik}^{(r)} > 0$ for $1 \leq k \leq n$:

$$\beta_i^{(r)} = \left[X^T D_i X\right]^{-1} X^T D_i Y \tag{3}$$

where X denotes the matrix in $\mathbb{R}^{n \times s}$ having x_k as its kth row. Y denotes the vector in $\mathbb{R}^n$ having y_k as its kth component, and D_i denotes the diagonal matrix in $\mathbb{R}^{n \times n}$ having $(U_{ik}^{(r)})^m$ as its kth diagonal element.
3) **Step 3.** The aim of this step is to update $U^{(r)} \rightarrow U^{(r+1)} \in M_f$, interpreting U_{ik} as the importance or weight attached to the extent to which the model value $f_i(x_k; \beta_i)$ matches y_k (fuzzy membership on all c models). The update is performed by the next formula:

$$U_{ik} = \left[\sum_{j=1}^{c} \left(\frac{E_{ik}}{E_{jk}}\right)^{\frac{1}{m-1}}\right]^{-1} \quad \text{if } E_{ik} > 0 \text{ for } 1 \leq i \leq c \tag{4}$$

In case we encounter some $E_{ik} = 0$, its value can be replaced by adding a small positive number (we used 10^{-100} in our experiments), so step 3 can be performed anyway.
4) **Step 4.** This step checks the termination of the algorithm. If the difference between U^r and U^{r+1} corresponding to two consecutive iterations is greater than the termination threshold, or r is less or equal to r_{max} then $r := r + 1$ and go to step 2. Otherwise stop.

A. Using fuzzy c-regression to generate synthetic data

Once we have introduced all the proper concepts relative to our work, the next step is to combine fuzzy clustering and switching regression models to generate synthetic data. In the previous section we have pointed out the formulas we use to implement the Fuzzy c-Regression models and now we present the basic algorithm to generate the synthetic data (Algorithm 1).

III. Data utility

As we have stated in the introduction, we want to analyze the relationship between the similarity of the clustering structures obtained when doing clustering with the original and synthetic data, and the information loss incurred when releasing the synthetic data instead of the original data set. The aim of information loss measures is to assess the validity of the synthetic data for posterior analysis. In fact, it is expected that the results of any analysis using the perturbed data are similar to the results of the same analysis using the original data.

Specifically, the FCRM synthetic data generator produces new synthetic data with an associated information loss inversely proportional to the number of cluster representatives,

Algorithm 1: Using FCRM to generate synthetic data

Data: $S = \{(x_1, y_1), \ldots, (x_n, y_n)\}$, m, ϵ, r_{max}
Result: $S = \left\{(x_1, y_1^{'}), \ldots, (x_n, y_n^{'})\right\}$
begin
  Get initial partition $U^{(0)}$ (e.g., Fuzzy c-Means);
  $r := 1$;
  while $\left|U^r - U^{r+1}\right| > \epsilon$ **and** $r < r_{max}$ **do**
    Update the values for the c model parameters $\beta_i = \beta_i^{(r)}$ and then the measure of error $E_{ik}(\beta_i)$ applying Equations 2 and 3;
    Update the fuzzy membership on all c clusters applying Equation 4;
    $r := r + 1$;
  foreach $(x_s, y_s) \in S$ **do**
    $j := arg\ max_{i=1}^{c} U_{is}$;
    $y_s^{'} := f_i(x_s; \beta_j)$;

which corresponds to the parameter c of the FCRM, see Figure 1. This property of the FCRM synthetic data generator has been assessed in a previous work [5], and in this paper we want to assess whether the larger the number of clusters, the similarity between the clustering structures obtained from the original and the synthetic data proportionally increases.

Furthermore, the information loss measures naturally depend on the analyses to be performed. Due to this, some general information loss measures have been developed. The Probabilistic Information Loss (PIL) [20] is a widely used information loss measure that compares some basic statistics for the original and the perturbed (i.e., synthetic) data set.

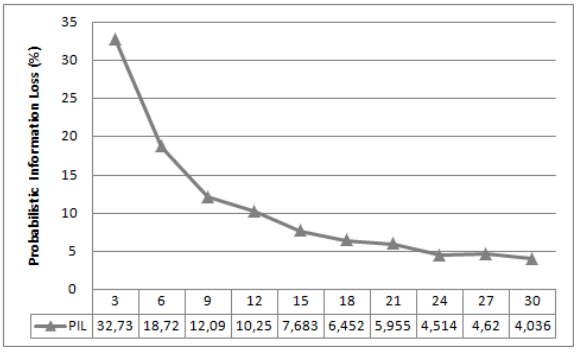

Fig. 1. Probabilistic Information Loss (PIL) when FCRM generates synthetic data for different values of c (horizontal axis).

IV. Clustering Methods analyzed

To evaluate the synthetic data generated by FCRM with respect to clustering we have compared the clustering structures obtained from the original data and the ones obtained from the synthetic data. Therefore, we need to apply some clustering methods, either crisp or fuzzy. In case of crisp methods we have applied the c-means (CM), and in case of fuzzy methods we have considered the fuzzy c-means (FCM), the noise clustering (NC) and the fuzzy possibilistic c-means (FPCM). We proceed now to describe each one of the mentioned clustering methods.

- **c-means (CM).** The c-means algorithm was first proposed by Stuart Lloyd in 1957 but later published in a journal in 1982 [19]. The c-means is one of the simplest unsupervised learning algorithms. The procedure follows a simple and easy way to classify a given data set through a certain number of clusters (assume c clusters) fixed a priori. Every iteration of this algorithm is composed of the following steps: (i) place c points into the space represented by the objects that are being clustered. These points represent initial group centroids, (ii) assign each object to the group that has the closest centroid and (iii) when all objects have been assigned, recalculate the positions of the c centroids. A loop has been generated. As a result of this loop we may notice that the c centroids change their location step by step until no more changes are done, i.e. the algorithm converges. Although it can be proved that the procedure will always terminate, the c-means algorithm does not necessarily find the most optimal configuration, corresponding to the global objective function minimum. The algorithm is also significantly sensitive to the initial randomly selected cluster representatives. The c-means algorithm can be run multiple times to reduce this effect.
- **Fuzzy c-means (FCM).** The fuzzy c-means algorithm [2] is one of the most widely used methods in fuzzy clustering. It is based on the concept of fuzzy c-partition, introduced by Ruspini in 1969 [25]. Furthermore, the FCM can be seen as the fuzzified version of the c-means algorithm, i.e. FCM allows one piece of data to belong to two or more clusters. From a conceptual point of view, the underlying data categories are considered as fuzzy. Then, with a set of objects $X = \{x_1, x_2, \ldots, x_N\}$ evaluated in terms of attributes $A = \{A_1, A_2, \ldots, A_M\}$ fuzzy c-means makes a fuzzy partition of the objects X. Therefore, considering c categories ($C = \{C_1, \ldots, C_c\}$) the problem turns out to be the determination of c membership functions $\mu_1, \mu_2, \ldots, \mu_c$, where μ_i is the membership function corresponding to C_i. μ_i are such that for each object x their membership to all category C adds to one.
  The FCM algorithm is also defined in terms of a function to minimize, and a solution is found by iterating over a loop similar to the one for the c-means. In this case, the function to minimize also considers a parameter m, which is the degree of fuzziness. The larger the m, the fuzzier the clusters. Specifically, when $m = 1$, the output of the algorithm is a crisp solution that corresponds to the c-means. Again, the fuzzy c-means procedure does not necessarily find the most optimal configuration.
- **Noise Clustering (NC)**. The noise clustering algorithm was first introduced in [8]. This method is based on FCM but introduces the concept of a *noise cluster* and

defines a similarity measure for this noise cluster. Accordingly, the NC algorithm defines an additional cluster that will collect the noisy data points with the special property that it is always at the same distance from every point in the data set. Therefore, this clustering technique reduces the effects of noisy data in the clusters obtained by FCM.

- **Fuzzy Possibilistic c-means (FPCM)** This clustering model is an extension of the possibilistic c-means (PCM) [16], which solves the noise sensitivity defect of FCM, and overcomes the coincident clusters problem of the PCM due to its sensitivity to good initializations. The FPCM [23] algorithm was proposed by N.R.Pal, K.Pal, and J.C.Bezdek, and it includes both the possibility(typicality) concept from PCM, and the membership concept from FCM. Hence, membership can be interpreted as a relative typicality that measures the degree to which a point belongs to one cluster relative to other clusters and is used to crisply label a data point. In addition, the possibility can be viewed as absolute typicality and it measures the degree to which a point belongs to one cluster taking into account all other data points. In this way, possibility can be used to reduce the effect of outliers. Altogether, combining both membership and possibility we obtain better clustering results.

V. Comparing Clustering Structures

After applying the clustering methods to either the original and synthetic data, we want to compute the similarity between both cluster structures. Even though a large number of evaluation criteria and similarity indexes for clustering structures have been proposed in the literature, we have just considered some of them. In case of crisp clustering we have taken into account the Rand and the Jaccard Index, whereas the Fuzzy Rand index and the α-cuts distance where considered in case of fuzzy clustering.

- **Rand Index (RI).** The Rand index [24] is a well-known measure of similarity between two crisp partitions of a data set. Let $\mathbf{P} = \{P_1, \cdots, P_k\} \subset 2^X$ and $\mathbf{Q} = \{Q_1, \cdots, Q_t\} \subset 2^X$ be two crisp partitions of a finite set $X = \{x_1, x_2, \cdots, x_n,\}$ with n elements, which means that $p_i \neq \emptyset$, $P_i \cap P_j = \emptyset$ for all $1 \leq i \neq j \leq k$, and $P_1 \cup P_2 \cup \cdots \cup P_k = X$ (and analogously for $\mathbf{Q}$). Let $C = \{(x_i, x_j) \in X \times X | 1 \leq i < j \leq n\}$ denote the set of all tuples of elements in X^2. We say that two elements $(x, x') \in C$ are *paired* in $\mathbf{P}$ if they belong to the same cluster, i.e., if there is a cluster $P_i \in \mathbf{P}$ such that $x \in P_i$ and $x' \in P_i$. Moreover, we distinguish the following subsets of C:
  - $C_1 \equiv$ the set of tuples $(x, x') \in C$ that are paired in $\mathbf{P}$ and paired in $\mathbf{Q}$.
  - $C_2 \equiv$ the set of tuples $(x, x') \in C$ that are paired in $\mathbf{P}$ but not paired in $\mathbf{Q}$.
  - $C_3 \equiv$ the set of tuples $(x, x') \in C$ that are not paired in $\mathbf{P}$ but paired in $\mathbf{Q}$.
  - $C_4 \equiv$ the set of tuples $(x, x') \in C$ that are neither paired in $\mathbf{P}$ nor in $\mathbf{Q}$.

  Obviously, C_1, C_2, C_3, C_4 is a partition of C, and $a+b+c+d = |C| = n(n-1)/2$, where $a = |C_1|$, $b = |C_2|$, $c = |C_3|$, and $d = |C_4|$. The tuples $(x, x') \in C_1 \cup C_4$ are the *concordant* pairs, i.e., the pairs for which there is agreement between $\mathbf{P}$ and $\mathbf{Q}$, while the tuples $(x, x') \in C_2 \cup C_3$ are the *discordant* pairs for which the two partitions disagree. The Rand index is then defined by the number of concordant pairs divided by the total number of pairs:
  $$RI(\mathbf{P}, \mathbf{Q}) = \frac{a+d}{a+b+c+d}$$
  Thus defined, the Rand index is a similarity measure which takes values between 0 and 1, where 1 means maximum similarity, i.e., $\mathbf{P} = \mathbf{Q}$, and consequently, 0 means maximum dissimilarity.
- **Jaccard Index (JI).** The Jaccard index, also known as the Jaccard similarity coefficient, is a statistic used for comparing the similarity and diversity of sample sets. The Jaccard index measures similarity between two crisp partitions of a data set, and is defined as follows:
  $$JI(\mathbf{P}, \mathbf{Q}) = \frac{a}{a+b+c}$$
- **Fuzzy Rand Index (FRI).** The Fuzzy Rand index [15] is a fuzzy variant of the Rand index which is able to compare any pair of fuzzy partitions. Given a fuzzy partition $\mathbf{P} = P_1, P_2 \cdots P_k$ of X, each element $x \in X$ can be characterized by its membership vector $\mathbf{P}(x) = (P_1(x), P_2(x) \cdots P_k(x)) \in [0,1]^k$, where $P_i(x)$ is the degree of membership of x to the ith cluster P_i. There exists a fuzzy equivalence relation on X in terms of a similarity measure on the associated membership vectors. Generally, this relation is of the form
  $$E_P(x, x') = 1 - \|\mathbf{P}(x) - \mathbf{P}(x')\|$$
  where $\|\cdot\|$ is a proper distance on $[0,1]^k$. The basic requirement on this distance is that it yields values in $[0,1]$. Now, a pair (x, x') is considered as being concordant in so far as $\mathbf{P}$ and $\mathbf{Q}$ agree on their degree of equivalence. Then the *degree of concordance* is defined as
  $$1 - |E_{\mathbf{P}}(x, x') - E_{\mathbf{Q}}(x, x')| \in [0,1]$$
  Analogously, the *degree of discordance* is
  $$|E_{\mathbf{P}}(x, x') - E_{\mathbf{Q}}(x, x')|$$
  Finally, the distance measure on fuzzy partitions is defined by the normalized sum of degrees of discordance:
  $$d(\mathbf{P}, \mathbf{Q}) = \frac{\sum_{(x,x') \in C} |E_{\mathbf{P}}(x, x') - E_{\mathbf{Q}}(x, x')|}{n(n-1)/2}|$$
  Likewise,
  $$1 - d(\mathbf{P}, \mathbf{Q})$$
  corresponds to the normalized degree of concordance. Hence, it is a generalization of the Rand index for fuzzy partitions.
- **α-cuts.** To compare the fuzzy clustering structures we have considered, in addition to the FRI, the α-cuts distance. In this distance, all those elements with a membership value larger than α have their membership

assigned to the value 1. Then, we have computed the absolute distance between memberships. In case of binary memberships, the α-cuts distance corresponds to the Hamming distance. This distance has been used before to compare clustering structures in [17].

VI. Experiments

We have organized the experiments considering the crisp clustering and the fuzzy clustering separately. In both cases the analysis is done by constructing pairs of files of the form (*original*, *synthetic*), and then computing the clustering on each file. Once each pair of clustering structures are built (i.e., crisp or fuzzy partitions), we obtain the clustering similarities by computing the different indexes and distances mentioned in Section V. Moreover, this process is repeated twenty times and we obtain the average clustering similarities.

Furthermore, we have considered two original files named as **orig4** and **orig9**. These files have been extracted from a test data [3] used in the European project CASC. We refer to the "Census" dataset which contains 1080 records with 13 numerical attributes labeled from $v1$ to $v13$. This dataset was used in the CASC project and in several other papers [7], [9], [11], [12], [10], [18], [27]. In **orig9** there are 9 dependent variables $v1, v3, v4, v6, v7, v9, v11, v12, v13$, and 3 independent variables, $v2, v8, v10$, while in **orig4** there are 4 dependent variables $v4, v7, v12, v13$, and 9 independent variables, $v1, v2, v3, v5, v6, v8, v9, v10, v11$.

Each one of the above original files has been perturbed by creating new synthetic data with two different synthetic data generators, the IPSO-A, IPSO-B, and IPSO-C methods [4] and the FCRM. Although this paper focuses on the FCRM synthetic data generator, we also considered the IPSO family of methods because, as stated in [17], the use of IPSO to generate synthetic data is appropriate when the user plans to apply clustering algorithms to the data. It is for this reason that we compare along these experiments the results obtained when generating synthetic data either with IPSO or FCRM.

Applying the IPSO family of methods to the **orig4** original file we obtained three protected files named as **anon4a**, **anon4b**, and **anon4c** corresponding to the synthetic data generated with IPSO-A, IPSO-B, and IPSO-C respectively. In the same way, applying the IPSO family of methods to the **orig9** original file we obtained three more protected files named as **anon9a**, **anon9b**, and **anon9c**.

On the other hand, we have protected both the **orig4**, and **orig9** original files by generating new synthetic data with FCRM. In this case we have run the FCRM with different values of its parameters m and c. In case of m, the degree of fuzziness, we have considered two possible values, $m_1 = 1.5$, and $m_2 = 2$, whilst the possible values for c, the number of cluster representatives, are $c = 3 \cdot i, i = 1, \cdots, 10$. Therefore, we have protected each one of the original files using FCRM as many times as the Cartesian product $m \times c$. This protected files will be referenced within this section as **mV**$_{-}$(**4**|**9**) for $V = 1, 2$.

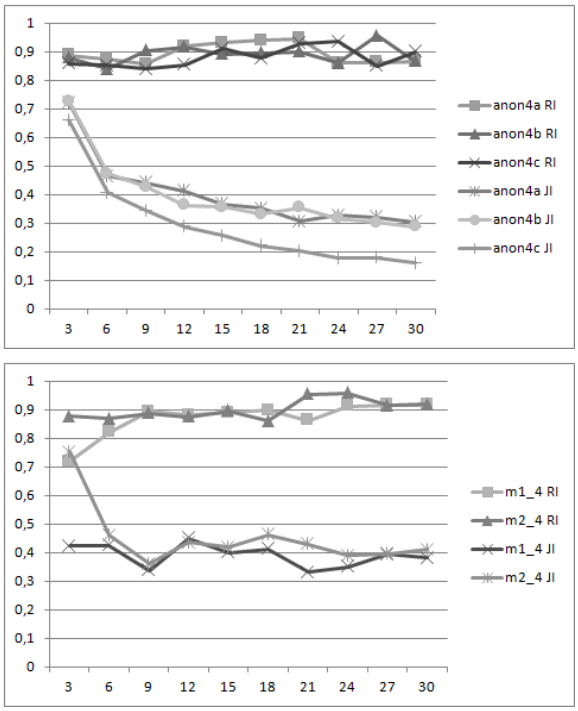

Fig. 2. Indexes computed from the clustering structures of **c-means** for different values of c (horizontal axis). Rand index (**RI**) and Jaccard index (**JI**) for the **orig4** original file and protection methods IPSO-A, IPSO-B and IPSO-C (top) and FCRM (bottom) with m_1 and m_2.

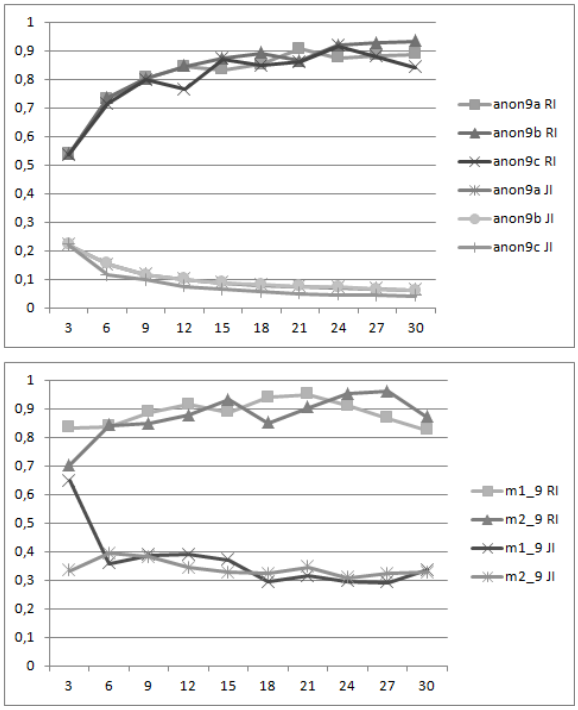

Fig. 3. Indexes computed from the clustering structures of **c-means** for different values of c (horizontal axis). Rand index (**RI**) and Jaccard index (**JI**) for the **orig9** original file and protection methods IPSO-A, IPSO-B and IPSO-C (top) and FCRM (bottom) with m_1 and m_2.

A. Performing crisp clustering

In case of crisp clustering, we have built the clustering structures by means of c-means bootstrapping it with a maximum number of iterations of 30 and a termination threshold of 0.00001. If we consider separately the cases where there are 4 and 9 dependent variables, we obtain the Figure 2 when evaluating the synthetic data generated with the three variants of IPSO, and the Figure 3 when evaluating the synthetic data generated with FCRM for different degrees of fuzziness (m_1 and m_2). These figures show that the FCRM and the IPSO synthetic data generators obtain similar clusters, so we have that the FCRM behaves with respect to the crisp clustering almost as good as IPSO does.

B. Performing fuzzy clustering

In case of fuzzy clustering, we have built the clustering structures by means of three different clustering methods, the FCM, NC and FPCM. All of them have been parameterized with a maximum number of iterations of 30 and a termination threshold of 0.00001. In addition to the previous parameters, in case of FPCM $\eta = 0.5$. To assess the similarity between the fuzzy partitions obtained by the previous clustering methods, we have computed the Fuzzy Rand index (FRI) and the α-cuts distance for c in $\{3, 6, 9, 12, 15, 18, 24, 27, 30\}$.

In case of FCM, we have protected both the *orig4* and *orig9* original files with the IPSO family of methods and the FCRM synthetic data generator. Figure 4 shows the α-cuts distances for all values of α in $\{0.2, 0.4, 0.6, 0.8\}$ when protecting orig4 with IPSO-C and FCRM with m_2. In the same way, the Figure 5 shows the α-cuts distances when protecting the orig9 original file. In both cases, the m parameter of FCM was fixed to 2. These figures reflect the similarity between the results obtained when protecting with IPSO-C or FCRM. There is just a slightly difference when $c = 3$, in this case the distance is a bit higher when protecting orig4 with IPSO-C than FCRM. However, when protecting orig9, the higher distance is obtained by FCRM.

On the other hand, Figures 6 and 7 show the FRI computed when protecting orig4 and orig9, respectively. We have used the same parameters as in the previous experiment but considering also m_1 when protecting with FCRM and in this case the m parameter of FCM was fixed to 1.5. Although both figures show some differences in the FRI value for small values of c, both IPSO and FCRM converge to FRI values around 0.8. Nevertheless, FCRM reaches higher FRI values than IPSO, in both the orig4 and orig9 cases. Hence, FCRM keeps the clustering properties, at least, in the same way as IPSO.

In case of NC, we have built the clustering structures from the orig4 and orig9 original files and the corresponding protected files for $m = 1.5$ and c in $\{3, 6, 9, 12, 15, 18, 24, 27, 30\}$. Specifically, the protection has been done by IPSO-C and FCRM with $m = 1.5$. Figures 8 and 9 show the α-cuts distances when considering 4 or 9 dependent variables, orig4 and orig9, respectively. Following the same rationale of the previous experiments

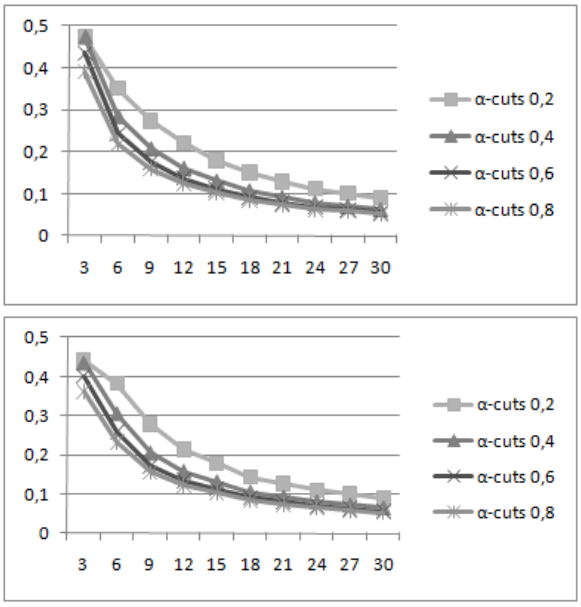

Fig. 4. α**-cuts** distances for fuzzy clusters obtained by **Fuzzy c-means** from the original file **orig4** for m_2 and different values of c (horizontal axis). Data protected using IPSO-C (top) and FCRM with m_2 (bottom).

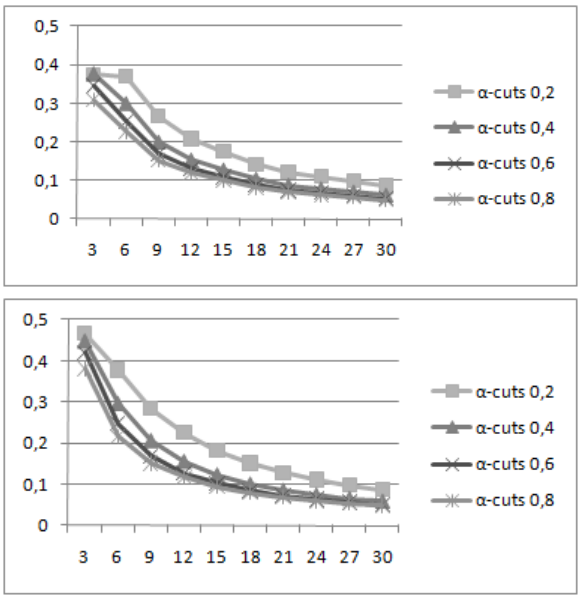

Fig. 5. α**-cuts** distances for fuzzy clusters obtained by **Fuzzy c-means** from the original file **orig9** for m_2 and different values of c (horizontal axis). Data protected using IPSO-C (top) and FCRM with m_2 (bottom).

with FCM, these figures show a similar α-cuts distance curve when the protection has been done by IPSO-C or FCRM. Again the higher differences are found when $c < 9$, but both synthetic data generators converge to distances in between 0.5 and 0.1. These results show that, the synthetic data generated by FCRM, as well as when it is generated by IPSO-C, is appropriate when clustering algorithms are planned to be applied to the data.

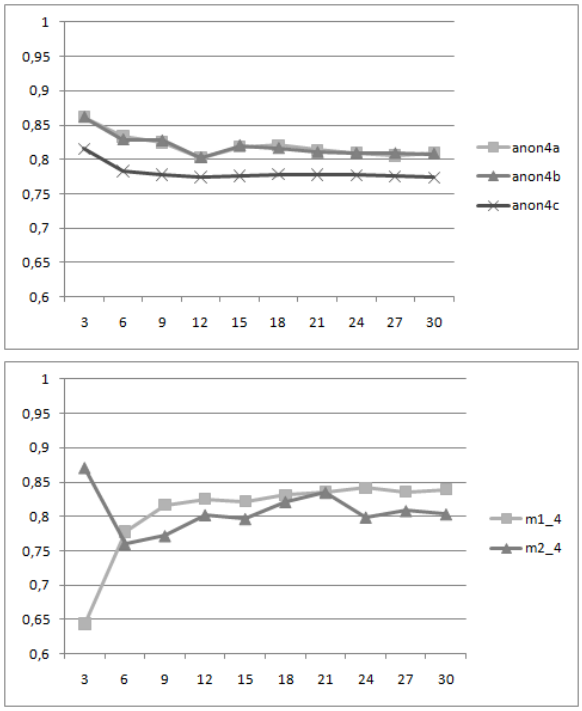

Fig. 6. **Fuzzy Rand index** for fuzzy clusters obtained by **Fuzzy c-means** from the original file **orig4** for m_1 and different values of c (horizontal axis). Data protected using IPSO-A, IPSO-B and IPSO-C (top) and FCRM (bottom) with m_1 and m_2.

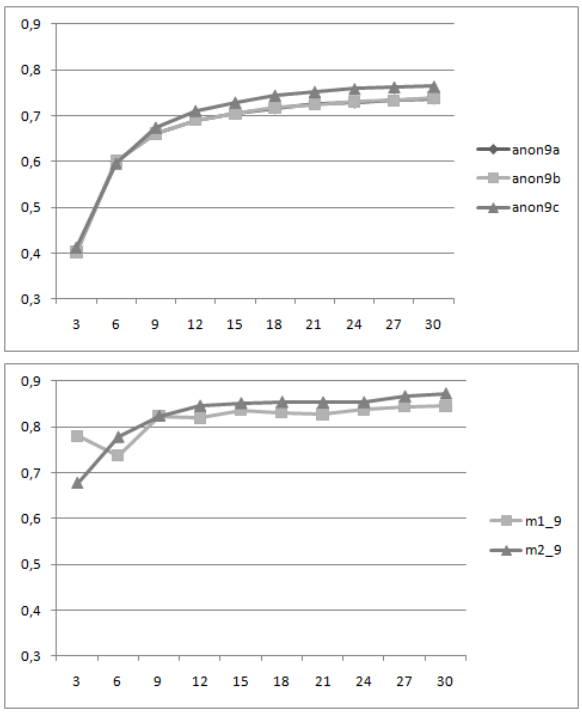

Fig. 7. **Fuzzy Rand index** for fuzzy clusters obtained by **Fuzzy c-means** from the original file **orig9** for m_1 and different values of c (horizontal axis). Data protected using IPSO-A, IPSO-B and IPSO-C (top) and FCRM (bottom) with m_1 and m_2.

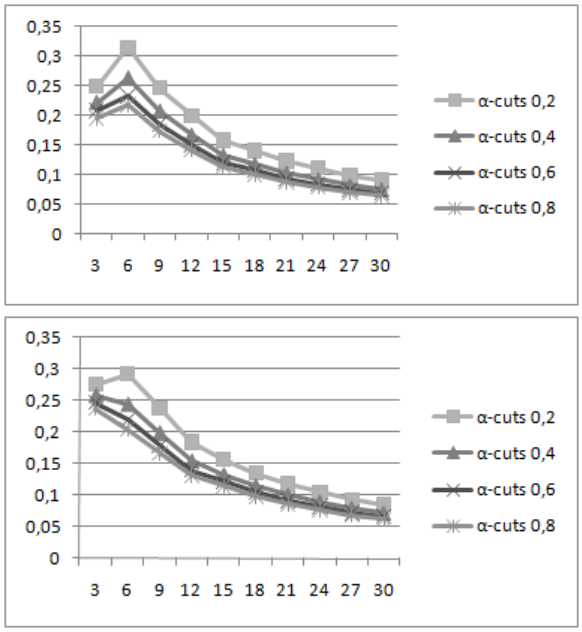

Fig. 8. α**-cuts** distances for fuzzy clusters obtained by **Noise clustering** from the original file **orig4** for m_1 and different values of c (horizontal axis). Data protected using IPSO-C (top) and FCRM with m_1 (bottom).

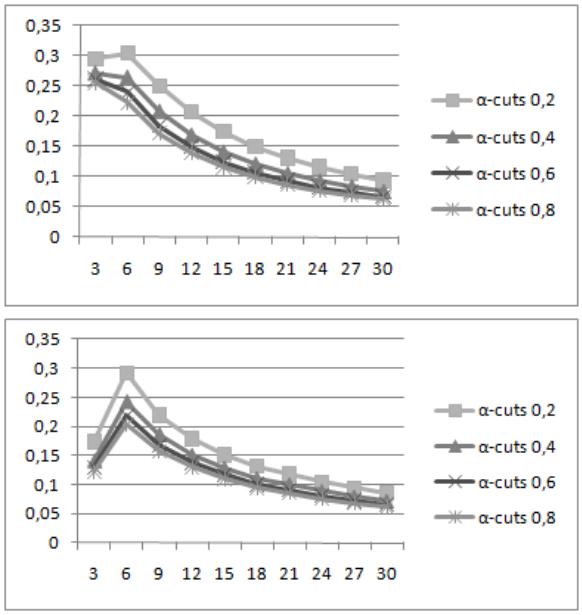

Fig. 9. α**-cuts** distances for fuzzy clusters obtained by **Noise clustering** from the original file **orig9** for m_1 and different values of c (horizontal axis). Data protected using IPSO-C (top) and FCRM with m_1 (bottom).

Finally, Figure 10 aims to compare the FRI's curve shapes for the different fuzzy clustering algorithms considered in this paper (i.e., FCM, NC and FPCM). The clustering similarity for FCRM when $c = 3$ is worse than IPSO-C, but for larger c values FCRM obtains slightly better similarities than IPSO-C in all the three cases. Therefore, FCRM is appropriate when clustering either with FCM, NC or FPCM. In this experiment we have considered orig4 as the original

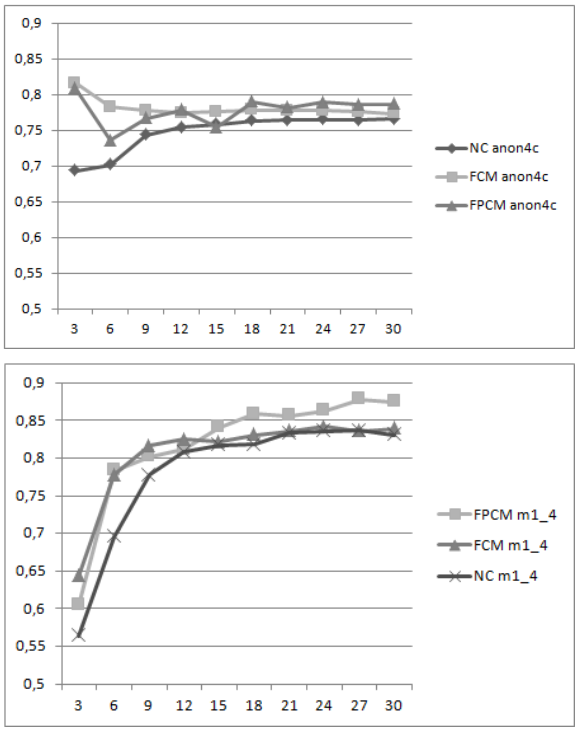

Fig. 10. **Fuzzy Rand index** values from the clustering structures of **Fuzzy c-means** (FCM), **Noise clustering** (NC) and **Fuzzy Possibilistic c-means** for m_1 and different values of c (horizontal axis). Data protected using IPSO-C (top) and FCRM with m_1 (bottom).

file and we have generated the synthetic data with IPSO-C and FCRM with $m = 1.5$.

VII. Conclusions

In this paper we have evaluated the FCRM synthetic data generator with respect to crisp and fuzzy clustering. We conclude that the generation of synthetic data by FCRM is as appropriate as it is when generating synthetic data with the IPSO family of methods and this protected data is planned to be studied or modeled with either crisp or fuzzy clustering algorithms.

In addition, we can derive from the results that the FRI clustering similarity measure is inversely proportional to the probabilistic information loss incurred when replacing the original data set with the synthetic data generated by FCRM. Similarly, the α-cuts clustering similarity, as it is a distance, is directly proportional to the probabilistic information loss.

Acknowledgments

This work is partially supported by the Spanish MEC (CONSOLIDER INGENIO 2010 CSD2007-00004, and TSI2007-65406-C03-02).

References

[1] Aggarwal, C.C, Yu, P.S., (2008) Privacy Preserving Data Mining: Models and Algorithms, Springer.

[2] Bezdek, J., (1981) Pattern Recognition with Fuzzy Objective Function Algorithms, Plenum Press, New York.

[3] Brand, R., Domingo-Ferrer, J., Mateo-Sanz, J.M., (2002) Reference data sets to test and compare SDC methods for protection of numerical microdata. European Project IST-2000-25069 CASC, http://neon.vb.cbs.nl/casc

[4] Burridge, J. (2003) Information Preserving Statistical Obfuscation. Statistics and Computing 13:321-327.

[5] Cano, I., Torra, V., (2009) Generation of Synthetic Data by Means of Fuzzy c-Regression. Proceedings of the 2009 IEEE International Conference on Fuzzy Systems.

[6] C. Clifton, D. Marks, Security and privacy implications of data mining, Proceedings of the ACM SIGMOD Workshop on Research Issues on Data Mining and Knowledge Discovery, pp. 15-19, 1996.

[7] Dandekar, R., Domingo-Ferrer, J., Sebé, F., (2002) Lhs-based hybrid microdata vs rank swapping and microaggregation for numeric microdata protection. In J. Domingo-Ferrer, editor, Inference Control in Statistical Databases, volume 2316:153-162 of Lecture Notes in Computer Science, Berlin Heidelberg, Springer.

[8] Dav, R.N., (1991) Characterization and detection of noise in clustering. Pattern Recognition Letters, Vol. 12:657-664.

[9] Domingo-Ferrer, J., Mateo-Sanz, J.M., Torra, V., (2001) Comparing SDC methods methods for microdata on the basis of information loss and disclosure risk. In Pre-proceedings of ETK-NTTS'2001, Vol. 2:807-826, Luxemburg. Eurostat.

[10] Domingo-Ferrer, J., Torra, V., Mateo-Sanz, J.M., Sebé, F., (2006) Empirical Disclosure risk assessment of the ipso synthetic data generators. In Monographs in Official Statistics-Work Session On Statistical Data Confidenciality, pages 227-238, Luxemburg. Eurostat.

[11] Domingo-Ferrer, J., Sebé, F., Solanas, A., (2005) A polynomial-time approximation to optimal multivariate microaggregation. Computers and Mathematics with Applications, Vol. 55(4):714-732.

[12] Domingo-Ferrer, J., Torra, V., (2005) Ordinal, continuous and heterogeneous k-anonymity through microaggregation. Data Mining and Knowledge Discovery, Vol. 11(2):195-212.

[13] Domingo-Ferrer, J., Torra, V., (2001) A quantitative comparison of disclosure control methods for microdata, Confidentiality, disclosure, and data access : Theory and practical applications for statistical agencies. Doyle, P.; Lane, J.I.; Theeuwes, J.J.M.; Zayatz, L.V. eds., Elsevier, pp. 111-133.

[14] Hathaway, R.J., Bezdek, J.C., (1993) Switching Regression Models and Fuzzy Clustering, IEEE Transactions on Fuzzy Systems, Vol. 1(3):195-204.

[15] Hüllermeier, E., Rifqi, M., (2009) A Fuzzy Variant of the Rand Index for Comparing Clustering Structures. Proceedings of the IFSA-EUSFLAT 2009, pp. 1294-1298.

[16] Krishnapuran, R., Keller., J.M., (1993) A possibilistic approach to clustering. IEEE Transactions on Fuzzy Systems. Vol. 1:98-100.

[17] Ladra, S., Torra, V., (2010) Information loss for synthetic data through fuzzy clusterin. International Journal of Uncertainty, Fuzziness and Knowledge-Based Systems. Vol. 18(1):25-37.

[18] Laszlo, M., Mukherjee, S., (2005) Minimum spanning tree partitioning algorithm for microaggregation. IEEE Transactions on Knowledge and Data Engineering. Vol. 17(7):902-911.

[19] Lloyd., S. P., (1982) Least squares quantization in PCM, IEEE Transactions on Information Theory. Vol. 28(2):129-137.

[20] Mateo-Sanz, J.M., Domingo-Ferrer, J., Sebé, F. Probabilistic information loss measures in confidentiality protection of continuous microdata, Data Mining and Knowledge Discovery, Vol. 11, pp. 181-193. Sep 2005. ISSN: 1384-5810

[21] Muralidhar, K., Sarathy, R., (2003) A theoretical basis for perturbation methods. Statistics and Computing 13:329-335.

[22] Muralidhar, K., Sarathy, R., (2008) Generating Sufficiency-based Non-Synthetic Perturbed Data. Transactions on Data Privacy Vol. 1(1):17-33.

[23] Pal, N.R., Pal, K., Bezdek, J.C., (1997) A mixed c-means clustering model. In IEEE Int.Conf.Fuzzy Systems, p. 11-21.

[24] Rand, W.M., (1971) Objective criteria for the evaluation of clustering methods. Journal of the American Statistical Association, Vol. 66(336):846-850.

[25] Ruspini, E., (1969) A new approach to clustering, Information and Control, Vol. 15:22-32.

[26] Willenborg, L., De Waal, T., (2001) Elements of Statistical Disclosure Control, New York: Springer-Verlag.

[27] Yancey, W.E., Winkler, W.E., Creecy, R.H., (2002) Disclosure risk assessment in perturbative microdata protection. In J. Domingo-Ferrer, editor, Inference Control in Statistical Databases, Vol. 2316:195-152 of Lecture Notes in Computer Science, Berlin Heidelberg, Springer.