Attribute Selection in Multivariate Microaggregation

Jordi Nin Javier Herranz Vicenç Torra
IIIA, Artificial Intelligence Research Institute
CSIC, Spanish National Research Council
Campus UAB s/n, 08193 Bellaterra,
Catalonia, Spain
{jnin, jherranz, vtorra}@iiia.csic.es

ABSTRACT

Microaggregation is one of the most employed microdata protection methods. The idea is to build clusters of at least k original records, and then replace them with the centroid of the cluster. When the number of attributes of the dataset is large, a common practice is to split the dataset into smaller blocks of attributes. Microaggregation is successively and independently applied to each block. In this way, the effect of the noise introduced by microaggregation is reduced, but at the cost of losing the k-anonymity property.

The goal of this work is to show that, besides of the specific microaggregation method employed, the value of the parameter k, and the number of blocks in which the dataset is split, there exists another factor which can influence the quality of the microaggregation: the way in which the attributes are grouped to form the blocks. When correlated attributes are grouped in the same block, the statistical utility of the protected dataset is higher. In contrast, when correlated attributes are dispersed into different blocks, the achieved anonymity is higher, and, so, the disclosure risk is lower. We present quantitative evaluations of such statements based on different experiments on real datasets.

Categories and Subject Descriptors

K.4 [**Computers and Society**]: Privacy; H [**Information Systems**]

General Terms

Microaggregation, Attribute Selection, Statistical Disclosure Control

1. INTRODUCTION

It is a common case that some confidential data has to be released to third parties (e.g., politicians and researchers) for data analysis. This dissemination has to be in accordance with laws and regulations to avoid the dissemination of confidential information. That is, published data has to preserve the privacy of the respondents.

There are two main approaches to privacy. The cryptographic and the perturbative approach. The perturbative one consists on distorting the original data file so that the resulting data does not permit the disclosure of sensitive information. Such distorted data is the one released. A large number of protection methods exist (see e.g. [4] and [9]). Perturbative data protection methods, besides of protecting the privacy of the respondents, should preserve *as much as possible* the statistical utility of the original data. As privacy and statistical utility are in contradiction, the problem can be seen as about finding a method with a good trade-off between privacy and statistical utility.

Recently, microaggregation has emerged as one of the most promising protection methods. For example, [13] shows that microaggregation is used by many statistical agencies for data anonymization.

The basic implementation of microaggregation works as follows: given a dataset with V attributes, it builds small clusters of at least k elements (records) and replaces the original records by the centroid of the cluster to which the records belong. A certain level of privacy is ensured because k records have an identical protected value (*k-anonymity*). Note that there are other ways to achieve k-anonymity; in some ones (just as it happens with basic microaggregation), the released dataset enjoys k-anonymity as a whole (see [2], for example). In other solutions, the data holder chooses different subsets of attributes, and k-anonymity is ensured, independently, for each of these subsets of attributes (see [14]).

When the number V of attributes is large, however, the basic microaggregation technique suffers from a low statistical utility (see for example [1]), especially if the attributes are not much correlated. This is so because in this case the distances between original records in the dataset and the centroids are quite large. Therefore, much information on the original data is lost and not included in the released (protected) dataset.

To solve this drawback, the following natural strategy is applied by statistical agencies: the dataset is split into smaller blocks of attributes, and microaggregation is applied independently to each block. In this way, the information loss is

lower but at the cost of a loss in the achieved level of privacy. Indeed, the property of k-anonymity is not ensured now. For example, the k records which fall in the same cluster for the first block of attributes, can fall in different clusters for all the other blocks of attributes. So, the resulting protected records will not be equal and no k-anonymity is ensured.

Summing up, there are many factors which influence the final result of applying microaggregation to a dataset: the value of the parameter k, the specific microaggregation method (we consider in this paper MDAV [10] and two projection-based microaggregation methods [20]), the number of blocks into which the dataset is split (and the number of attributes in each block). In addition to these ones, there is another factor which should also be taken into account and that, up to our knowledge, has not been studied in detail before: how to select which attributes will form each block.

In this work we study this aspect in detail, and we show that the results (statistical utility, and privacy/anonymity levels) of applying microaggregation to a dataset can significantly vary according to the grouping strategy. We concentrate on two grouping strategies. The first one, widely accepted by statistical agencies, is focused on the maximization of the statistical utility. That is, (highly) correlated attributes are grouped in the same block(s) so that the distance between the original elements and the protected ones is small. The second strategy, which we propose here, consists of scattering the groups of correlated attributes into different blocks. This strategy is defined with the goal of obtaining *correlated blocks* so that a higher level of anonymity can be maintained. For example, when two records are in the same cluster for one block, and the blocks of attributes (as a whole) are *correlated* to each other, then it is possible with some probability that these two records also fall in the same cluster for all the other blocks. This would lead to two identical protected records. In other words, the idea of this new strategy, is to enjoy some anonymity (higher privacy) even in the case in which attributes are microaggregated by blocks (higher data utility).

We have tested these two strategies with real datasets. In order to see the differences between the two strategies more clearly, we have chosen datasets with strong correlations between some of the attributes. The results of the experiments support our intuitions: the first strategy leads to a lower information loss, but it is more vulnerable to privacy attacks; the second strategy suffers from a higher information loss, but it maintains a higher level of anonymity, and so the disclosure risk is lower. The consequence is that one strategy or the other can be followed, depending on the scenario and on the importance given to data utility and privacy.

1.1 Organization of the paper

In Section 2 we review some basic concepts related to protection methods, in general (and microaggregation in particular). We discuss in Section 3 the two main strategies to group attributes before applying microaggregation to the resulting blocks, and the intuitive differences between them, in terms of the expected properties (data utility, privacy) of the resulting microaggregation. We give a simple example to illustrate this intuition. Section 4 is devoted to test the two strategies with real datasets; we explain the ingredients of our experiments and the obtained results. Finally, we list in Section 5 the consequences that we can extract from this work, concerning both the technique of microaggregation by blocks in general and the particular strategies to select the attributes which form each block.

2. PRELIMINARIES

In this section we explain some basic concepts useful in the rest of the paper. Namely, we first describe the scenario where a microdata protection method is applied to preserve the privacy of the owners of some statistical data. Then we recall one of the most used protection methods, microaggregation, and some of its variants. Finally, we describe some ways, to measure the quality of a microdata protection method, according to the levels of privacy and statistical utility that it provides.

2.1 Statistical Data Protection

A dataset X can be seen as a matrix with n rows (*records*) and V columns (*attributes*), where each row contains V attributes of an individual. The attributes in a dataset can be classified in two different categories, *identifiers* or *quasi-identifiers*, depending on their capability to identify unique individuals. Among the quasi-identifier attributes, we distinguish between *confidential* and *non-confidential*, depending on the kind of information they contain. Because of this, we write $X = X_{id}||X_{nc}||X_c$.

In this paper, we consider the following scenario for statistical disclosure control, which was defined in [9] to compare several protection methods.

(i) Identifier attributes in X are either removed or encrypted. Therefore we write $X = X_{nc}||X_c$.

(ii) Confidential quasi-identifier attributes X_c are not modified, and so we have $X'_c = X_c$; in this way, the statistical utility of the confidential attributes is completely preserved.

(iii) A microdata protection method ρ is applied to non-confidential quasi-identifier attributes, in order to preserve the privacy of the individuals whose confidential data is being released. This leads to a protected dataset $X'_{nc} = \rho(X_{nc})$.

(iv) The released dataset is $X' = X'_{nc}||X'_c = \rho(X_{nc})||X_c$.

2.2 Microaggregation

Microaggregation is one of the most popular, studied and used microdata protection methods. It builds small clusters of at least k elements of v attributes and replaces the original records by the centroid of the cluster to which the records belong.

The goal of a microaggregation method is to minimize the total Sum of Square Error

$$SSE = \sum_{i=1}^{c} \sum_{x_{ij} \in C_i} (x_{ij} - \bar{x}_i)^T (x_{ij} - \bar{x}_i),$$

where c is the total number of clusters, C_i is the i-th cluster and $\bar{x}_i$ is the centroid of C_i. The restriction is $|C_i| \geq k$, for all $i = 1, \ldots, c$.

If a microaggregation method is applied to all the V attributes of the original dataset X at the same time; then, the resulting protected dataset X' satisfies the property of k-anonymity: each protected record can correspond to at least k original records. However, in order to increase the statistical utility of the released (protected) information, statistical agencies usually split the whole dataset X in blocks of a few attributes, and then apply a microaggregation method to each block, independently. In this way, k-anonymity is not preserved any more.

In the case of univariate microaggregation ($v = 1$), there exist polynomial time algorithms to obtain the optimal microaggregation [15]. However, for the multivariate case ($v > 1$), the problem of finding the optimal microaggregation is NP-hard. For this reason, multivariate microaggregation methods are heuristic. We recall here two of these multivariate techniques.

2.2.1 Projection-based Microaggregation

Previous references to this idea [8] considered only the particular case where all the V attributes of the dataset are projected. But a more general case can also be implemented and analyzed: microaggregation is applied in parallel to blocks of v_i attributes, where $\sum_i v_i = V$ (See [20] for a complete description).

Formally, a projection-based microaggregation algorithm, when applied to a dataset X with n records and V attributes, works as follows:

- Split the dataset X into r sub-datasets $\{X_i\}_{1 \leq i \leq r}$, each one with v_i attributes and n records, such that $\sum_{i=1}^{r} v_i = V$.
- For each sub-dataset X_i:
  1. Apply a projection algorithm to the attributes in X_i. This results in an univariate vector z_i with n components (one for each record).
  2. Sort the components of z_i in increasing order.
  3. Apply, to the sorted vector z_i, an univariate optimal microaggregation method (for instance, the method defined in [15]). The unique difference with the standard univariate case is the cost (SSE) function. Here, univariate methods have to take into account the SSE values of the unprojected data.
  4. For each cluster resulting from the previous step, compute the v_i-dimensional centroid and replace all the records in the cluster by the centroid.

Depending on the projection method used on the attributes, we will obtain different methods of multivariate microaggregation. Due to the fact that they should preserve as much statistical properties of the data as possible (as this is desirable in the scenario of statistical data protection), the PCP and Zscores projection methods [3, 6, 7] are one of the best choices. We call the resulting microaggregation algorithms *PCP-microaggregation* and *Zscore-microaggregation*.

2.2.2 MDAV Microaggregation

The MDAV algorithm [10, 16] is an heuristic algorithm for clustering records in a dataset X so that all the clusters except one (the last one) contain exactly k records. The algorithm is as follows.

Algorithm (MDAV) (X: dataset, k: integer) is

1. while ($|X| > k$) do
  (1.a) Compute the average record $\bar{x}$ of all records in X.
  (1.b) Consider the most distant record x_r to the average record $\bar{x}$.
  (1.c) Form a cluster around x_r. The cluster contains x_r together with the $k-1$ closest records to x_r. Replace all the records in this cluster by the average record of the cluster.
  (1.d) Remove these records from dataset X.
2. if ($|X| > k$) then
  (2.a) Find the most distant record x_s from record x_r (from step 1.b)
  (2.b) Form a cluster around x_s. The cluster contains x_s together with the $k-1$ closest records to x_s. Replace all the records in this cluster by the average record of the cluster.
  (2.c) Remove these records from dataset X.
3. end if
4. end while
5. Form a cluster with the remaining records

The *most distant record* and the *closest records* are usually computed using the Euclidean distance, and *the average record* is defined as the arithmetic mean of the records.

2.3 Measures to Evaluate Risk and Utility

A microdata protection method must guarantee a certain level of privacy (low disclosure risk). At the same time, since the goal is to allow third parties to perform reliable statistical computations over the released (protected) data, the protection method must ensure somehow that the protected data is statistically close to the original data.

Therefore, we have two inversely related aspects to measure for each microdata protection method: the *disclosure risk* (DR), which is the risk that an intruder obtains correct links between the protected and the original data; and the *information loss* (IL) caused by the protection method. When one of them increases, the other one decreases. The two extreme cases are the following ones: (i) if the original microdata is released, then information loss is zero, but the disclosure risk is maximal; (ii) if the original microdata is encrypted and then released, the disclosure risk is zero (if we exclude the possibility that the protected attributes are strongly statistically related to other known unprotected attributes), but the information loss is maximal.

2.3.1 Generic measures

There are different generic measures proposed in the literature to evaluate the quality of a data protection method. We will use the *score*, which was introduced in [8] and used in several papers [19, 22, 23] to compare protection methods. The score is a simple and natural way to evaluate the trade-off between the information loss and the disclosure risk because it is defined as the average of these two values. Namely,

$$\text{score} = \frac{(IL + DR)}{2},$$

where information loss and disclosure risk are computed as follows.

- **Information Loss (IL).** The overall IL is computed as $IL = 100\big(0.2\,IL_1 + 0.2\,IL_2 + 0.2\,IL_3 + 0.2\,IL_4 + 0.2\,IL_5\big)$, where

  (i) IL_1 is the mean absolute error of the original microdata X with respect to the protected data X'.

  (ii) IL_2 is the mean variation of the attribute average vectors.

  (iii) IL_3 is the mean variation of the attribute covariance matrices.

  (iv) IL_4 is the mean variation of the attribute variance vectors.

  (v) IL_5 is the mean variation of the attribute correlation matrices.

- **Disclosure Risk (DR).** To compute this measure, one considers two different approaches, the first one being the *interval disclosure risk*, ID, which is the average percentage of protected values falling into the intervals around their corresponding original values. The second approach is *record linkage risk*, which considers the scenario where an intruder has obtained an original record $x \in X$, possibly from a different external dataset Y, and tries to link it with the corresponding protected record $x' \in X'$. If he succeeds, then he can match the non-protected confidential information x_{nc} with the identifiers x_{id} that he obtained from Y, and so he breaks the privacy of this individual. Two standard record linkage methods are considered:

  - Distance-based record linkage [21], where the original record is linked to the closest protected record, using for example the Euclidean distance. The average percentage of correctly linked records using this method is the *Distance based Linkage Disclosure risk*, DLD.
  - Probabilistic record linkage [17], where the link is assigned in a probabilistic way, according to some criterion on some coincidence vectors (defined from the available sets of original and protected records). The average percentage of correctly linked records using this method is the *Probabilistic Linkage Disclosure risk*, PLD.

  When computing disclosure risk, half weight is given to record linkage and half weight is given to interval disclosure. Then, the risk of record linkage is defined as the average of the two methods. Formally, this corresponds to $DR = 0.25 \cdot DLD + 0.25 \cdot PLD + 0.5 \cdot ID$.

2.3.2 Other quality measures for microaggregation

Some microdata protection methods admit other measures to evaluate their quality. This is the case of microaggregation, whose goal is to minimize the total Sum of Square Error SSE. Since there are no optimal solutions in polynomial time to multivariate microaggregation, and the methods used are heuristic, the actual value of SSE for a given method is a measure of its quality.

Regarding privacy, microaggregation provides, by definition, some level of anonymity. If the method is applied to all the attributes (a single block), then the initial parameter k indicates the achieved anonymity: there are at least k protected records which are identical. However, if the original dataset is split into r blocks and the microaggregation method is applied to each block independently, then the final level of anonymity obviously decreases: two records which fall in the same cluster in one block may fall in different clusters in other blocks, which results in different protected records. Actually, in the (unrealistic) case where all the entries of the dataset X are random and independent, it is easy to see that the expected number of original records which are mapped to the same protected record is

$$n \cdot \left(\frac{k}{n}\right)^r, \tag{1}$$

where n is the number of original records, k is the initial anonymity parameter, and r is the number of blocks.

In realistic cases, a way of computing the real level of anonymity achieved by a microaggregation method is to consider the ratio between the total number n of records and the number of protected records which are different. This gives the average size of each "global cluster" in the protected dataset. We denote as k' this *real anonymity* measure,

$$k' = \frac{n}{|\{x'|x' \in X'\}|}.$$

This measure k' tells us the average cardinality of the global clusters. For example, with the chosen measure k', a released database with 5 different clusters, all of them with cardinality 2, and another database with 5 clusters as well, one of them with cardinality 6, and the rest with cardinality 1, both lead to $k' = 2$. But the achieved levels of security are different: in the first database, all the records are equally protected, whereas in the second one, there are 6 highly protected records, and 4 records which are unique. It would be possible to consider other definitions for this measure k', taking into account the variance of the cardinalities of all the global clusters, in order to distinguish the average case from the worst case, for example.

3. HOW TO GROUP ATTRIBUTES TO BE MICROAGGREGATED

To apply microaggregation to a dataset X, we need to settle the method itself (i.e., which variation we will apply), the

parameter k, and the number of blocks the dataset X is split into. However, these are not the only parameters to be considered when the number of blocks r is larger than 1. In this case, the way in which the attributes are grouped into blocks affects in an important way the results and the quality of the microaggregation.

It is standard practice to select the attributes on the basis of statistical utility. It is clear that if highly correlated attributes are considered, objects similar with respect to one attribute will be similar with respect to another one. Due to this, if microaggregation is applied to correlated attributes, clusters will contain objects that are similar with respect to all the attributes included in the cluster. Therefore, this approach results into microaggregation with low information loss.

Nevertheless, as usual, statistical utility and privacy are contradictory terms. The experiments in Section 4 show that, as expected, the disclosure risk of microaggregation in this case is higher than when correlated attributes are put into different blocks.

More specifically, we also study in Section 4 a different approach. Blocks are formed in such a way that first attributes of each block are (highly) correlated, second attributes of each block are (highly) correlated, and so on. In some way, we construct "correlated blocks", instead of constructing blocks with correlated attributes. The goal of this new approach is to try to increase the resulting real anonymity k'. If two records are in the same cluster of the same block, this means that the first attributes of these records in this block are more or less close to each other, and the same for the second attributes of this block, etc. Then, when we move to another block, if the j-th attribute of this new block is correlated with the j-th attribute of the previous block, it is likely that the attributes of this second block corresponding to these two records will be again close to each other, and so they will fall in the same cluster, again, with some non-negligible probability. Ideally, many records will fall inside the same clusters, for each block of attributes, and so the number of protected records which will be exactly equal will be higher, increasing in this way the real anonymity and the privacy level of the released dataset. Of course, the probability of maintaining a good level of anonymity decreases very quickly when the number r of blocks is high (remember the unrealistic but orientating formula for the expected size of the global clusters, stated in equation (1), at the end of previous Section 2.3). But for small values of r, the difference between the two types of grouping strategies, in terms of the achieved real anonymity k', is appreciable, as we will see in our experiments, in Section 4.

Before moving to these experiments involving real datasets, we want to illustrate the arguments explained above with a simple example, where the two grouping strategies are easy to distinguish and lead to very different results. In general, this will not be the case with real datasets, where it is not always easy to find enough (high) correlations between attributes, and so the differences between applying one grouping strategy or another may be slight.

3.1 A Simple Example

Let X be a dataset, to be protected via microaggregation, which contains information on 4 attributes of 16 individuals (i.e. the number of records is $n = 16$ and the number of attributes is $V = 4$). The complete dataset is presented in Table 1.Original. Assume that the first and second attributes are highly correlated (they are equal, actually), and the same happens with the third and fourth attributes, while the correlation factor between the first or second attributes and the third or fourth attributes is very low. In this case, if we want to independently microaggregate two blocks of two attributes each, the two grouping strategies are clearly distinguishable.

In the first one, we group the first and the second attributes on the one hand, and the third and fourth attributes on the other hand. We have applied the MDAV algorithm with $k = 2$. In this case, we have obtained a very low SSE value (equal to 0), therefore we can ensure that the information loss is very low. However, the protected dataset has the real anonymity equal to 1. Then an intruder is able to link all the protected records with the original ones. Therefore, the protected dataset obtained using this attribute selection has a very high disclosure risk (low real anonymity). If we compute the score, the measure explained in Section 2.3, we obtain a Score=50 because IL=0 and DR=100. The protected data is presented in Table 1 (correlated columns).

Following the second strategy, we group the first and the third attributes on the one hand, and the second and fourth attributes on the other hand. We have applied the same microaggregation algorithm with the same parameterization than in the former case (MDAV with $k = 2$). Now, the obtained SSE is equal to 5.98, and therefore, the information loss is higher than in the first case. However, the real anonymity (k') is equal to 2, and so, the disclosure risk is lower than in the case of correlated selection. Now, if we compute the score, we obtain a surprising result: the IL is equal to 22.68, the DR is equal 43.18 and the final Score is equal 32.93. So, it is clear that the score is much lower than in the correlated case. In other words, the trade-off between IL and DR is much more in favour of the non-correlated case than of the correlated one. The protected data is presented in Table 1 (non-correlated columns)

4. EXPERIMENTS

We have tested the two different strategies for attribute grouping with real data extracted from two large datasets available in the Internet. The first one, denoted as Water-treatment dataset, was extracted from the UCI repository [18] and contains 35 attributes corresponding to 380 entries or records. The second dataset, called EIA, was extracted from the U.S. Energy Information Authority [12], and has been used as a reference dataset in many works dealing with statistical data protection, e.g. [5, 9, 19]. It contains 4092 records and 10 attributes.

We have reduced both datasets, to have only 9 attributes, with the idea of forming 3 blocks of 3 attributes each. In this scenario it is easier to apply and compare the two attribute grouping strategies. Namely, if attributes $a1, a2, a3$ are highly correlated with each other, and the same happens for $a4, a5, a6$ on the one hand, and $a7, a8, a9$ on the

	Correlated					Non-correlated				
	IL	DLD	PLD	ID	Score	IL	DLD	PLD	ID	Score
Mic.MDAV05	14.14	73.03	67.24	72.73	42.79	31.75	8.16	39.87	45.79	33.32
Mic.MDAV10	18.78	61.97	55.66	63.56	39.98	28.28	5.26	28.95	43.00	29.16
Mic.MDAV15	17.34	49.74	43.95	56.99	34.63	35.60	2.50	18.82	41.89	30.94
Mic.MDAV20	18.28	39.34	35.53	51.18	31.29	32.44	2.63	14.21	39.34	28.16
Mic.MDAV25	21.68	32.37	29.08	48.59	30.67	36.74	1.71	12.89	30.85	27.91
Mic.PCP05	18.36	40.39	30.39	60.82	33.23	50.41	8.95	2.11	36.63	35.74
Mic.PCP10	18.11	30.00	21.58	53.66	28.92	53.51	5.00	0.79	30.96	35.22
Mic.PCP15	21.67	23.82	20.39	50.54	29.00	56.28	4.21	1.32	30.37	36.42
Mic.PCP20	25.17	21.45	16.05	47.21	29.08	61.02	4.74	1.05	26.30	37.81
Mic.PCP25	23.25	19.08	13.68	49.34	28.05	62.48	3.82	0.26	25.61	38.15
Mic.Zscore05	17.62	76.05	62.50	68.65	43.29	98.33	10.13	3.16	42.96	61.57
Mic.Zscore10	20.62	63.82	54.87	61.53	40.53	108.75	6.05	2.24	40.61	65.57
Mic.Zscore15	20.99	54.08	47.76	56.42	37.33	113.74	5.39	1.71	40.18	67.80
Mic.Zscore20	20.74	47.76	40.79	53.48	34.81	114.78	3.03	1.45	39.98	67.94
Mic.Zscore25	24.30	43.95	34.47	54.04	35.46	113.71	3.29	1.05	37.60	66.80

Table 2: Scores of different microaggregation methods and parameterizations using the Water-treatment dataset. Mic.Method$k$ corresponds to microaggregation using method Method (MDAV, PCP or Zscore) with initial anonymity value k

Original				Correlated				Non-correlated			
a_1	a_2	a_3	a_4	a'_1	a'_2	a'_3	a'_4	a'_1	a'_2	a'_3	a'_4
1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.5	1.0	1.5	1.0
1.0	1.0	8.0	8.0	1.0	1.0	8.0	8.0	3.0	5.5	3.0	5.5
2.0	2.0	1.0	1.0	2.0	2.0	1.0	1.0	1.5	1.0	1.5	1.0
2.0	2.0	8.0	8.0	2.0	2.0	8.0	8.0	4.0	5.5	4.0	5.5
3.0	3.0	2.0	2.0	3.0	3.0	2.0	2.0	3.5	2.0	3.5	2.0
3.0	3.0	7.0	7.0	3.0	3.0	7.0	7.0	3.5	7.0	3.5	7.0
4.0	4.0	2.0	2.0	4.0	4.0	2.0	2.0	3.5	2.0	3.5	2.0
4.0	4.0	7.0	7.0	4.0	4.0	7.0	7.0	3.5	7.0	3.5	7.0
5.0	5.0	3.0	3.0	5.0	5.0	3.0	3.0	3.0	5.5	3.0	5.5
5.0	5.0	6.0	6.0	5.0	5.0	6.0	6.0	5.5	6.0	5.5	6.0
6.0	6.0	3.0	3.0	6.0	6.0	3.0	3.0	4.0	5.5	4.0	5.5
6.0	6.0	6.0	6.0	6.0	6.0	6.0	6.0	5.5	6.0	5.5	6.0
7.0	7.0	4.0	4.0	7.0	7.0	4.0	4.0	7.0	4.5	7.0	4.5
7.0	7.0	5.0	5.0	7.0	7.0	5.0	5.0	7.0	4.5	7.0	4.5
8.0	8.0	4.0	4.0	8.0	8.0	4.0	4.0	8.0	4.5	8.0	4.5
8.0	8.0	5.0	5.0	8.0	8.0	5.0	5.0	8.0	4.5	8.0	4.5

Table 1: Example of MDAV Microaggregation with correlated attributes $((a_1, a_2)(a_3, a_4))$ and non-correlated attributes $((a_1, a_3)(a_2, a_4))$.

other hand, the first strategy (correlated attributes) will lead to blocks $(a1, a2, a3)$, $(a4, a5, a6)$ and $(a7, a8, a9)$, whereas the second strategy (correlated blocks) will lead to blocks $(a1, a4, a7)$, $(a2, a5, a8)$ and $(a3, a6, a9)$.

In the case of the Water-treatment dataset, there are many attributes (and possible groups) of highly correlated attributes. We have chosen the following ones:

- a1. Input pH to plant (PH-E)
- a2. Input pH to primary settler (PH-P)
- a3. Input pH to secondary settler (PH-D)
- a4. Input chemical demand of oxygen to plant (DQO-E)
- a5. Input conductivity to primary settler (COND-P)
- a6. Input conductivity to secondary settler (COND-D)
- a7. Output Biological demand of oxygen (DBO-S)
- a8. Output suspended solids (SS-S)
- a9. Output sediments (SED-S)

In the case of the EIA dataset we have chosen the following attributes:

- a1. Unique utility identification number (UTILITYID)
- a2. Utility name (UTILNAME)
- a3. Reporting year for the data (YEAR)
- a4. Sales to residential consumers (RESSALES)
- a5. Revenue from sales to commercial consumers (COM-REVENUE)

- a6. Sales to commercial consumers (COMSALES)
- a7. Reporting month for the data (MONTH)
- a8. Revenue from sales to residential consumers (RESREVENUE)
- a9. Revenue from sales to industrial consumers (INDREVENUE)

Tables 2 to 7 summarize the results of the experiments. We have applied to each dataset three different microaggregation methods: MDAV, PCP-microaggregation and Zscores-microaggregation. For each dataset and each method, we have tested five different parameterizations according to the initial value of k ($k = 5, 10, 15, 20, 25$ for the Water-treatment dataset, and $k = 5, 25, 50, 75, 100$ for the EIA dataset). Finally, we have run all these experiments for the two considered attribute grouping strategies: correlated attributes, where blocks are $(a1, a2, a3)$, $(a4, a5, a6)$ and $(a7, a8, a9)$, and non-correlated attributes (which corresponds to "correlated blocks"), where blocks are $(a1, a4, a7)$, $(a2, a5, a8)$ and $(a3, a6, a9)$.

First we concentrate on the generic measures for the information loss and the disclosure risk (and so, the score). Table 2 shows the results obtained in the case of the Water-treatment dataset. The differences between the two strategies are very evident, because the first one leads to much lower values of the information loss, whereas the second one leads to much lower values of the disclosure risk. For instance, comparing the information loss of the Zscore microaggregation, correlated attributes selection obtains IL values between 17.62 and 24.30, whereas the non-correlated selection obtains values between 98.33 and 113.71. Regarding the three employed methods, MDAV has the best scores in the non-correlated scenario (27.91 is the best one, PCP and Zscore microaggregation always obtain scores over 35.00), whereas PCP-microaggregation has the best scores in the correlated case (28.05 is the best one). The behaviour of Zscores-microaggregation is quite surprising: it has quite good scores in the correlated case, but very bad scores (in particular, very high information loss) in the case of "correlated blocks".

Similar results are presented in Table 5, where the measures are computed for the EIA dataset. Here, the comparison between correlated and non-correlated results is not as different as in the Water-treatment dataset. In our opinion, this is so because the correlations among the attributes are not so high. However, if one observes the information loss values presented in this table, it is easy to see that IL values are lower in the correlated case. See, for instance, the IL values in the MDAV microaggregation for the correlated selection. They are between 6.68 and 17.63. In contrast, for the non-correlated ones, IL values are between 10.05 and 25.14.

Regarding disclosure risk, we observe that non-correlated selection presents lower disclosure risk than correlated one. For instance, if one observes the values for the distance based and probabilistic record linkage (DLD and PLD) and interval disclosure (ID) for the Mic.PCP05 configuration in the Water-treatment dataset, it is clear that correlated selection has higher disclosure risk than non-correlated selection. In particular, DLD, PLD and ID values for the correlated case are 40.39, 30.39, and 60.82 respectively, whereas in the non-correlated case DLD, PLD and ID values are 8.95, 2.11, and 36.63.

Now, we consider the performance measures for microaggregation, the values of SSE and the real anonymity k'. We consider different situations where an intruder can have access to one (the first one), two (the first two ones) or the three blocks of protected data. Tables 3 and 6 show the results for real anonymity k', whereas Tables 4 and 7 show the results for SSE.

Of course, if the intruder has access only to one block, then the real anonymity k' roughly coincides with the initial value of k. In fact, it is larger because for microaggregation with $k = K$ the number of records in a cluster is in the interval $[K, 2K)$. In the general case, the tables show that k' decreases rapidly with respect to the number of blocks considered. Also, as expected, k' is always larger when we consider correlated blocks. Note that the differences between the k' values of the two strategies are noticeable, especially, when only two blocks of attributes are considered, and when the initial anonymity value k is quite large. Furthermore, both strategies lead to higher values of k' than those which would be obtained in the "unrealistic" totally random case (recall formula (1) in Section 2.3). For example, if we consider the Water-treatment dataset (see Table 3) with two ($r = 2$) groups of attributes and $k = 25$, then the unrealistic case would lead to a real anonymity around $625/380 = 1.64$, but the two realistic strategies lead to values around $k' = 2.1$ (for the first strategy) and values between 2.16 and 4.87 (for the second strategy).

SSE behaves more or less as the information loss: it is lower when the initial value of k is small, and it is lower in the correlated case than in the non-correlated case. The three microaggregation methods obtain very similar results for the SSE, in both the correlated and non-correlated scenarios, so we cannot deduce from this experiment that any of them provides a better solution to the original microaggregation problem.

5. CONSEQUENCES

From the results obtained in the experiments, we can extract some consequences which are valid either for the microaggregation technique in general or for the specific strategies to group attributes in blocks. In this section we discuss these consequences.

The first of them is that the real anonymity that microaggregation provides, when the dataset is split into blocks of attributes, decreases very quickly when the number of blocks increases, independently of the strategies for grouping attributes. For example, for standard values of the initial parameter k, less than 25, we observe that real anonymity is almost non-existent if the number of blocks is $r = 3$ (or more). Therefore, if k'-anonymity was the main motivation to choose microaggregation as a data protection method, one should either start with a large value for the initial k, or split the dataset into only one or two blocks of attributes.

	Correlated					Non-correlated				
	IL	DLD	PLD	ID	Score	IL	DLD	PLD	ID	Score
Mic.MDAV05	6.68	1.78	2.87	87.60	25.82	10.05	1.61	2.43	83.30	26.36
Mic.MDAV25	11.83	0.78	0.68	79.55	25.99	16.58	0.81	0.56	72.27	26.53
Mic.MDAV50	13.23	0.60	0.56	72.37	24.85	21.86	0.62	0.49	67.16	27.86
Mic.MDAV75	15.56	0.53	0.55	72.16	25.95	20.26	0.68	0.60	64.61	26.44
Mic.MDAV100	17.63	0.39	0.49	67.52	25.81	25.14	0.60	0.48	61.29	28.02
Mic.PCP05	16.61	1.78	2.87	65.29	25.21	19.37	0.62	0.54	57.35	24.17
Mic.PCP25	18.33	0.78	0.68	61.62	24.76	22.07	0.63	0.48	53.45	24.54
Mic.PCP50	19.77	0.60	0.56	59.71	24.96	22.25	0.64	0.48	52.41	24.37
Mic.PCP75	21.16	0.53	0.55	59.67	25.63	22.70	0.64	0.49	52.11	24.52
Mic.PCP100	22.26	0.39	0.49	57.87	25.71	23.07	0.66	0.50	50.93	24.42
Mic.Zscore05	12.58	6.17	8.36	75.09	26.88	16.81	2.58	3.82	75.75	28.14
Mic.Zscore25	16.04	5.03	5.98	70.88	27.12	17.06	1.92	2.55	75.21	27.89
Mic.Zscore50	16.84	4.92	5.60	69.36	27.07	17.25	1.44	2.22	73.88	27.55
Mic.Zscore75	18.69	3.91	5.30	69.71	27.92	17.83	1.25	2.14	73.71	27.77
Mic.Zscore100	18.86	3.48	4.63	67.78	27.39	17.80	1.16	1.88	70.95	27.02

Table 5: Score of different microaggregation methods and parameterizations using the EIA dataset. Mic.Method$k$ corresponds to microaggregation using method Method (MDAV, PCP or Zscore) with initial anonymity value k

	Correlated			Non-correlated		
	1G	2G	3G	1G	2G	3G
Mic.MDAV05	5.28	1.02	1.00	5.00	1.16	1.01
Mic.MDAV10	10.00	1.15	1.01	10.00	1.84	1.13
Mic.MDAV15	15.20	1.38	1.01	15.20	2.66	1.38
Mic.MDAV20	20.00	1.64	1.03	20.00	3.76	1.50
Mic.MDAV25	25.33	2.11	1.09	25.33	4.87	1.87
Mic.PCP05	5.35	1.04	1.00	5.07	1.04	1.00
Mic.PCP10	10.00	1.11	1.00	10.00	1.18	1.01
Mic.PCP15	15.20	1.30	1.01	15.20	1.41	1.02
Mic.PCP20	20.00	1.66	1.03	20.00	1.78	1.06
Mic.PCP25	25.33	2.09	1.04	25.33	2.16	1.12
Mic.Zscore05	5.43	1.03	1.00	5.00	1.06	1.01
Mic.Zscore10	10.00	1.16	1.01	10.00	1.35	1.05
Mic.Zscore15	15.20	1.33	1.02	15.20	1.84	1.10
Mic.Zscore20	20.00	1.62	1.03	20.00	2.59	1.26
Mic.Zscore25	25.33	2.15	1.07	25.33	3.62	1.43

Table 3: Real k' values of different microaggregation methods and parameterizations for different number of groups known by the intruder using the Water-treatment dataset. Mic.Method$k$ corresponds to microaggregation using method Method (MDAV, PCP or Zscore) with initial anonymity value k

	Correlated	Non-correlated
Mic.MDAV05	28.18	69.51
Mic.MDAV10	46.14	126.21
Mic.MDAV15	72.03	173.96
Mic.MDAV20	94.24	259.07
Mic.MDAV25	114.56	247.58
Mic.PCP05	28.59	93.67
Mic.PCP10	49.61	133.83
Mic.PCP15	71.99	170.12
Mic.PCP20	91.96	206.74
Mic.PCP25	110.91	229.50
Mic.Zscore05	23.78	73.52
Mic.Zscore10	49.05	115.77
Mic.Zscore15	72.23	160.30
Mic.Zscore20	93.10	197.20
Mic.Zscore25	111.69	231.81

Table 4: SSE values of different microaggregation methods and parameterizations using the Water-treatment dataset. Mic.Method$k$ corresponds to microaggregation using method Method (MDAV, PCP or Zscore) with initial anonymity value k

	Correlated			Non-correlated		
	1G	2G	3G	1G	2G	3G
Mic.MDAV05	5.12	1.10	1.03	5.01	1.15	1.06
Mic.MDAV25	25.42	1.83	1.26	25.10	2.06	1.28
Mic.MDAV50	50.52	3.20	1.57	50.52	4.43	1.81
Mic.MDAV75	75.78	4.95	1.92	75.78	7.08	2.30
Mic.MDAV100	102.30	7.54	2.57	102.30	10.94	3.17
Mic.PCP05	5.27	1.02	1.00	5.14	1.02	1.01
Mic.PCP25	25.42	1.27	1.04	25.10	1.33	1.06
Mic.PCP50	50.52	1.91	1.15	50.52	2.16	1.20
Mic.PCP75	75.78	2.87	1.30	75.78	3.39	1.40
Mic.PCP100	102.30	4.28	1.53	102.30	5.00	1.73
Mic.Zscore05	5.18	1.02	1.01	5.08	1.03	1.01
Mic.Zscore25	25.26	1.44	1.06	25.10	1.65	1.12
Mic.Zscore50	50.52	2.59	1.28	50.52	3.45	1.52
Mic.Zscore75	75.78	4.45	1.58	75.78	6.13	2.18
Mic.Zscore100	102.30	6.90	2.02	102.30	9.79	3.16

Table 6: Real k' values of different microaggregation methods and parameterizations for different number of groups known by the intruder using the EIA dataset. Mic.Method$k$ corresponds to microaggregation using method Method (MDAV, PCP or Zscore) with initial anonymity value k

	Correlated	Non-correlated
Mic.MDAV05	28.74	45.18
Mic.MDAV25	145.36	212.44
Mic.MDAV50	219.45	361.38
Mic.MDAV75	313.31	468.30
Mic.MDAV100	397.48	569.66
Mic.PCP05	141.29	124.83
Mic.PCP25	203.81	251.99
Mic.PCP50	330.19	369.73
Mic.PCP75	469.74	482.05
Mic.PCP100	649.49	608.82
Mic.Zscore05	50.65	111.97
Mic.Zscore25	140.70	212.47
Mic.Zscore50	184.19	336.68
Mic.Zscore75	287.95	439.99
Mic.Zscore100	378.51	553.89

Table 7: SSE values of different microaggregation methods and parameterizations using the EIA dataset. Mic.Method$k$ corresponds to microaggregation using method Method (MDAV, PCP or Zscore) with initial anonymity value k

With respect to this, note, however, that microaggregation ranks among the best methods for data protection in [9] with respect to the trade-off between privacy and data utility. This is so, because even in the case that k-anonymity is not achieved, the perturbation added to the data might make re-identification difficult.

Then, if we focus on the overall evaluation of the method taking into account all measures for information loss, disclosure risk, SSE and real anonymity, obtained by the two strategies, it is very difficult to conclude that one of them is better than the other. As expected, when blocks are formed by correlated attributes, we obtain better results in terms of the information loss and SSE. On the contrary, for "correlated blocks", we obtain better results in real anonymity, and also in the disclosure risk. These aspects are more or less cancelled when computing the final score for each case: the scores obtained by the two strategies are very similar.

The clear consequence of this analysis is that the strategy for grouping attributes is another degree of freedom for microaggregation that has to be considered with care. As shown in the simple example of Section 3.1, it might be even possible to have much better results if we use blocks with uncorrelated attributes. A case that has not been reported in the literature.

Then, with real data, when choosing the value for k, one can take a small k if data utility is the main goal (at the cost of a lower level of privacy), and a larger k if privacy is the main concern. Analogously, one can microaggregate the whole dataset as a single block, if privacy is considered to be more important than data utility; or one can form a higher number of blocks, if data utility is the most desired property of the protection.

In the case of the grouping strategy selection, giving priority to data utility corresponds to choosing the first strategy, correlated attributes in the same block(s). This can be the case if the protected data is going to be released to a more or less reliable (or restricted) network. However, if the protected data is going to be widely released, for example in the Internet, then maybe privacy is considered to be the main concern; in this case, the second strategy, "correlated blocks", should be chosen, because it enjoys a higher anonymity level and a lower disclosure risk.

6. ACKNOWLEDGMENTS

Partial support by the Spanish MEC (projects ARES – CONSOLIDER INGENIO 2010 CSD2007-00004 – and eAEGIS – TSI2007-65406-C03-02) and Generalitat de Catalunya (grant 2005-SGR-00093) is acknowledged. Jordi Nin wants to thank the Spanish Council for Scientific Research (CSIC) for his I3P grant.

7. REFERENCES

[1] Aggarwal, C. C. (2005), On k-anonymity and the curse of dimensionality. Proceedings of the 31st International Conference on Very Large Data Bases. VLDB Endowment, 901-909.

[2] Aggarwal, G., Feder, T., Kenthapadi, K., Khuller, S., Panigrahy, R., Thomas, D., Zhu, A. (2006), Achieving

anonymity via clustering. Proceedings of the 25th ACM Symposium on PODS. ACM, 153-162.

[3] Anwar, M. N. (1993), Micro-aggregation - The small Aggregates Methods. Internal report. Luxembourg, Eurostat.

[4] Adam, N. R., Wortmann, J. C.,(1989), Security-control for statistical databases: a comparative study, ACM Computing Surveys, Volume: 21, 515-556.

[5] CASC: Computational Aspects of Statistical Confidentiality, European Project IST-2000-25069, http://neon.vb.cbs.nl/casc.

[6] Defays, D., Nanopoulos, Ph., (1993), The Small Aggregates Method. Proceedings of the 92 Symposium on Design and Analysis of Longitudinal Surveys. Ottawa, Statistics Canada.

[7] Defays, D., Anwar, M. N. (1995), Micro-aggregation: A Generic Method. Proceedings of the 94 International Seminar on Statistical Confidentiality. Luxembourg. Office for Official Publications of the European Communities.

[8] Domingo-Ferrer, J., Torra, V., (2001), Disclosure control methods and information loss for microdata, Pages 91-110 of [11].

[9] Domingo-Ferrer, J., Torra, V., (2001), A quantitative comparison of disclosure control methods for microdata, Pages 111-133 of [11].

[10] Domingo-Ferrer, J., Martínez-Ballesté, A., Mateo-Sanz, J.M., Sebé, F. (2006) Efficient multivariate data-oriented microaggregation, The VLDB Journal, 15, 355-369.

[11] Doyle, P., Lane, J., Theeuwes, J., Zayatz, L., eds. (2001), Confidentiality, disclosure, and data access: theory and practical applications for statistical agencies, Elsevier Science.

[12] U.S. Energy Information Authority, http://www.eia.doe.gov/cneaf/electricity /page/eia826.html

[13] Felsö, F., Theeuwes, J., Wagner, G., (2001) Disclosure Limitation in Use: Results of a Survey, Pages 17-42 of [11].

[14] Fung, B., Wang, K., Yu, P.S. (2005), Top-down specialization for information and privacy preservation. Proceedings of the 21st IEEE ICDE'05, IEEE Computer Society, 205-216.

[15] Hansen, S., Mukherjee, S. (2003) A Polynomial Algorithm for Optimal Univariate Microaggregation. Trans. on Kwnoledge and Data Engineering, 15:4 1043-1044.

[16] Hundepool, A., Van de Wetering, A., Ramaswamy, R., Franconi, L., Capobianchi, A., DeWolf, P.-P., Domingo-Ferrer, J., Torra, V., Brand, R., Giessing, S. (2003) μ-ARGUS version 3.2 Software and User's Manual. Statistics Netherlands, Voorburg NL, feb 2003.

[17] Jaro, M. A. (1989) Advances in Record Linkage Methodology as Applied to Matching the 1985 Census of Tampa, Florida. Journal of the American Statistical Society, 84:406 414-420.

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

[19] Nin, J., Herranz, J., Torra, V. (2008) Rethinking Rank Swapping to Decrease Disclosure Risk, Data & Knowledge Engineering, 64:1, 346-364.

[20] Nin, J., Herranz, J., Torra, V. (2008) On the Disclosure Risk of Multivariate Microaggregation, submitted.

[21] Pagliuca, D., Seri, G., (1999), Some results of individual ranking method on the system of enterprise accounts annual survey, Esprit SDC Project, Delivrable MI-3/D2.

[22] Winkler, W. E., (2004), Re-identification methods for masked microdata, Privacy in Statistical Databases 2004, Lecture Notes in Computer Science, Springer-Verlag, volume: 3050, 216-230.

[23] Yancey, W. E., Winkler, W. E., R. H. Creecy, (2002), Disclosure risk assessment in perturbative microdata protection, Inference Control in Statistical Databases: From Theory to Practice, Lecture Notes in Computer Science, Springer-Verlag, volume: 2316, 135-152.