Information Sciences 278 (2014) 344–356

Contents lists available at ScienceDirect

Information Sciences

journal homepage: www.elsevier.com/locate/ins

An evolutionary algorithm to enhance multivariate Post-Randomization Method (PRAM) protections

Jordi Marés [a,b], Vicenç Torra [b,*]

[a] UAB, Universitat Autònoma de Barcelona, Catalonia, Spain
[b] IIIA, Artificial Intelligence Research Institute, CSIC, Spanish Council of Scientific Research, Campus de la UAB, 08193 Bellaterra, Catalonia, Spain

ARTICLE INFO

Article history:
Received 25 March 2013
Received in revised form 26 October 2013
Accepted 8 March 2014
Available online 3 April 2014

Keywords:
Information privacy
Post-Randomization Method
PRAM
Disclosure control
Evolutionary algorithm

ABSTRACT

The amount of public statistical information available is growing and more accurate protection methods are needed in order to achieve data confidentiality. The Post-Randomization Method (PRAM) protection method was introduced in 1997 as a very powerful method for categorical microdata, but it is still not widely used. This method has a Markov matrix as a parameter. The main problem of the application of this method is that it is difficult to find a good Markov matrix that performs changes in the microdata file producing low loss of valuable information and low risk of disclosure of sensitive data.

In this paper we present a methodology that helps us to find a matrix to perform better protections. This is achieved by using an evolutionary algorithm with integrated Information Loss and Disclosure Risk measures. Experiments using three different datasets are also presented in order to empirically evaluate the application of this technique.

1. Introduction

Nowadays dissemination of information is continuously growing and, because of that, protecting the confidentiality of individuals has become more important. Information such as social security numbers, business names, or personal ID numbers must be protected because it can uniquely identify an individual and, if it is identified, all its confidential information is going to be disclosed. Furthermore, protecting this information alone to individual identifiers may be not enough because, as it is described in [18], the identity of individuals can be obtained by linking groups of records between different datasets resulting in an unauthorized disclosure of sensitive information. Some examples of attributes to be protected together as quasi-identifiers are zip code, gender, and birth date.

In addition, when applying Statistical Disclosure Control (SDC) methods, one has to deal with two competing goals: the microdata file has to be safe enough to guarantee the protection of individual respondents but at the same time the loss of information should not be too large. A discussion about this issue can be found in [9].

There have been different approaches for Privacy-Preserving Data Mining [1] and Statistical Disclosure Control [14,21] in order to achieve better protections to prevent attacks to the confidential information about individuals from the disseminated data. Protection methods can be classified into two generic categories: the first one is the group of methods that mask the original dataset generating a modified version which are called perturbative methods, and the second one is the group of methods that generate synthetic data that preserve some desired characteristics of the original which are called non-perturbative methods.

* Corresponding author. Tel.: +34 935 809 570.
E-mail addresses: jmares@iiia.csic.es (J. Marés), vtorra@iiia.csic.es (V. Torra).

http://dx.doi.org/10.1016/j.ins.2014.03.057
0020-0255/

The data inside datasets can be of two types: categorical or continuous. In our case, we focus on categorical data. The problem of categorical data over continuous data is that there are less actions to perform in the protection process because arithmetic operations are not allowed. Then the only actions allowed with categorical data are the exchange of categories by others that already exist, categories suppression, and generalizations of some categories into new ones. This lack of possible operations makes the protection a difficult task.

There exist several protection methods applicable to categorical data. Some of them are Microaggregation [19], Top Coding, Bottom Coding, Global Recoding [15], and Post Randomization Method (PRAM).

PRAM is a powerful protection method that is based on changing categories on the basis of a Markov matrix. PRAM generalizes Top Coding, Bottom Coding, and Global Recoding as with a proper Markov matrix PRAM reduces to the previous methods.

There have been some discussions about PRAM and its properties, see e.g. [11,22]. Nevertheless PRAM is still not very widely used in categorical microdata protection because of the difficulty of obtaining a good Markov matrix for good protection. For example, in [7], data protection using PRAM leads to the worst results when compared to other data protection methods.

There also exist some work on finding the best PRAM matrices like in [4] where the authors provide an analytical approach to find optimal PRAM matrices. They use Information Loss measures different than the ones we use and they rely on analytical methods to find the best solution. However, our approach relies on an evolutionary algorithm that evolves autonomously without taking into account analytical methods.

In this paper, we show how Information Loss and Disclosure Risk measures can be integrated within an evolutionary algorithm to seek new and enhanced PRAM Markov matrices in order to obtain better protections for categorical microdata making. This work is an extension of [16] where we presented an approach that consists of a pre-masking optimization based on an evolutionary algorithm applied to an initial population of one PRAM matrix. However, while in [16] we considered only an initial population of a single PRAM matrix performing univariate protections, here we consider a population of more than one PRAM matrix in order to be able to perform multivariate protections. In addition we present all brand new experimental results in this framework of multivariate protections and using also more datasets than in [16].

The remainder of the paper is organized as follows. Section 2 describes the PRAM protection method, Section 3 introduces our proposed algorithm, experimental results with three data sets are given in Section 4 and, finally, concluding remarks and possible future work are listed in Section 5.

2. The PRAM protection method

The Post Randomization Method (PRAM) was described as a method for disclosure protection of microdata, especially focused on categorical variables.

The idea of this method is to change the categories on some categorical variables for certain records to different ones according to prescribed Markov matrices. Each matrix contains the swapping probabilities for all the possible pairs of categories of a single variable. Although those matrices are the key point for the PRAM method, it is difficult to find a good matrix that leads to a good trade-off between data perturbation (Information Loss) and Disclosure Risk. The Markov approach makes PRAM very general, because it generalizes noise addition, data suppression, and data recoding.

The following subsections describe the kind of matrix used in this work and the analytical measures used to evaluate the PRAM protection performance.

2.1. PRAM matrix

In the discussion we understand p_{kl} as the probability of exchanging category k to category l, and then the following equality $\sum_{l=1}^{n} p_{kl} = 1$ is required.

The type of matrix used in this work is a full matrix with off-diagonal elements depending on the frequencies in the original microdata file which is used in [5]. Formally, the probability p_{kk} is constant for all k and its value is given as a parameter. However, the probability p_{kl} where $k \neq l$, is computed as follows

$$p_{kl} = \frac{(1 - p_{kk})\left(\sum_{i=1}^{n} T_{\xi}(i) - T_{\xi}(k) - T_{\xi}(l)\right)}{(n-2)\left(\sum_{i=1}^{n} T_{\xi}(i) - T_{\xi}(k)\right)} \tag{1}$$

where $T_{\xi}(i)$ is the frequency of the ith category in the original dataset. Using this approach, the biggest exchange probabilities are assigned to the categories with lower frequency in the original dataset in order to cause more confusion in the protected dataset.

2.2. Analytical measures

After protecting microdata, their quality needs to be assessed. To do that we used the two most popular measures: the Information Loss and the Disclosure Risk.

Information Loss [6] is known as the quantity of harm that is inflicted to the data by a given masking method. This measure is small when the analytic structure of the masked dataset is very similar to the structure of the original dataset, so, the motivation for preserving the structure of the dataset is to ensure that the masked dataset will be analytically valid and

interesting. In this work we used three measures of Information Loss that check different aspects on the data: contingency table-based information loss (CTBIL) [6], distance-based information loss (DBIL) [6], and entropy-based information loss (EBIL) [6]. In order to get a single indicator of the overall Information Loss we used the mean value of all three measures as shown in Eq. (2).

$$IL(X) = \frac{CTBIL(X) + DBIL(X) + EBIL(X)}{3} \tag{2}$$

Assessment of the protection method quality cannot be limited to Information Loss because Disclosure Risk has also to be measured. Disclosure Risk [7] is known as the amount of original information that can be obtained by an intruder from the masked dataset. This measure is small when the masked dataset values are different from the original ones. In this work we used three different Disclosure Risk measures. The first one is the interval disclosure (ID) [7], which checks the quantity of individual original values that can be discovered from the masked dataset. The second and third measures are the distance-based record linkage (DBRL) [8] and the probabilistic record linkage (PRL) [8] ant they check the number of masked records that can be linked to their original ones, this is, the number of disclosed records in the masked dataset. As in the Information Loss case we wanted to have a single Disclosure Risk value so we used the mean value of interval disclosure measure and the maximum value of record linkage measures as shown in Eq. (3).

$$DR(X) = \frac{ID(X) + max(DBRL(X), PRL(X))}{2} \tag{3}$$

Although the goal is to minimize both measures, they are inversely related. Because of this, if we perform a very aggressive protection we will obtain a very high Information Loss but a very low Disclosure Risk and, if we perform a very weak protection, we will obtain a very low Information Loss but a very high Disclosure Risk. Then, a protection will be considered good if it has low and balanced values for both measures.

3. Evolutionary algorithm for multivariate Post-Randomization Method (PRAM)

Evolutionary algorithms are stochastic processes inspired by biological evolution, generally oriented to find exact or approximate solutions to optimization or search problems.

Usually, evolutionary algorithms work with a population of solutions but they can also work taking into account the whole population as a single solution.

These algorithms maintain a population of individuals, denoted as $P(t)$ for generation t. Each individual $X'_j \in P(t)$ is evaluated by some measure of their "fitness". Fitness *evaluation* is used to guide individuals from generation to generation. Some of the selected members are *altered* by operators with an evolutive connotation, such as mutation and crossover. These operators create offspring from the existing population members from previous generations. Surviving individuals are evaluated again, and the process is repeated until some stopping criteria is reached. From this basic scheme many particularizations are possible [2,17].

Algorithm 1. Evolutionary Algorithm to Enhance PRAM Matrices

Input: $P(0) = \{X_1, \ldots, X_n\}$ initial population of PRAM matrices
Output: $P(t) = \{X'_1, \ldots, X'_n\}$ final population of PRAM matrices
$t \Leftarrow 0$
fitness_eval$(P(0))$
while *stopping*$(P(t)) \neq$ *true*; **do**
  $X_j \Leftarrow$ PRAM matrix selected randomly from the population
  alter $\Leftarrow$ randomly choose between mutation and cross
  if *alter* by mutation **then**
    $X'_j \Leftarrow$ *mutate*(X_j)
  else
    $X'_j \Leftarrow$ *cross*(X_j)
  end if
  if *fitness_eval*$(X'_j) <$ *fitness_eval*(X_j) **then**
    replace(X_j, X'_j)
  end if
  $t \Leftarrow t + 1$
end while
return $P(t)$

Algorithm 1 shows the pseudo-code of our algorithm, which is a generic evolutionary algorithm with some particularities.

The algorithm starts with an initial population that is a set of PRAM matrices (one matrix for each attribute to protect). Then, before the iterative process starts, we evaluate the fitness of this set of matrices when they are used to protect the original data. By doing the initial evaluation score we obtain a reference point to know whether the algorithm is improving or not at each generation. The iterative part, where each iteration represents a new generation, keeps altering the matrices using one of the two genetic operators: mutation and crossover. In order to not alter the values of matrices too much, and make the generations evolve more smoothly, the genetic operator to perform will be selected randomly with a probability of a 0.5 and only one of them will be applied in the same generation. After applying the operator to the selected matrix we obtain a new offspring and they have to be evaluated in order to know whether the new offspring is better than its parent or not (i.e. the offspring has lower fitness value than its parent). After fitness evaluation, an elitist replacement strategy is followed, which means that the new individual and its parent compete with each other and only the one with the best score (i.e. the lowest fitness score) will be selected as the individual in the population for the next generation. Finally, the stopping criteria of the iterative process is not fixed and it can be chosen by the user. Different examples of stopping criteria could be a maximum number of generations, to reach a certain fitness value, a minimum difference of fitness value between generations,...However, terminating the execution when the difference of fitness is below a certain threshold is dangerous as it is very likely to end too early because if none of the offspring is good enough to be inserted in the population, the fitness change for this generation will be zero but it does not mean that we reached an optimal solution.

In our case we have not used any of those and we let it run indefinitely and we stopped them manually when we ran out of time.

Another important issue to comment is that, as we are dealing with multivariate PRAM protections and we do not want to make abrupt changes into the PRAM matrices, only one matrix will be treated in each generation and its selection is done randomly.

Finally, at the end of our algorithm execution we obtain a set of PRAM matrices that perform better multivariate protections to the data than the original ones.

In the following subsections we will describe the key points of our evolutionary algorithm such as genotype encoding (Section 3.1), genetic operators (Section 3.2) and fitness function (Section 3.3), as well as the way we integrate the analytical measures presented in Section 2.2 within our evolutionary algorithm.

3.1. Genotype encoding

The individuals inside population should be represented in an easy and effective way to be dealt with by the algorithm. In our case, PRAM matrices are transition matrices based on probabilities so they contain numerical values with decimals. For the sake of simplicity we decided to multiply the values by 1000 and then take only the integer part. By doing this we will have integer values in the range $[0, 1000]$.

Although integer values are much easier to deal with than floating point values, it is still not a good representation. We decided then to convert the integer values into their binary representation. This representation is a good one to deal with when altering values because it only needs to flip bit values from 0 to 1 and vice versa. Finally, in [12] the authors showed that the use o Gray code in evolutionary algorithms allows to obtain faster and more accurate solutions than using regular binary representation so we decided to use this representation in our approach. A discussion about the use of Gray coding is also discussed in [3].

Fig. 1 shows the encoding process of a PRAM matrix into a Gray-coded genome matrix ready to be used in the evolutionary algorithm.

3.2. Genetic operators

We used the two most popular genetic operators in our approach: mutation and crossover [13].

Mutation is performed by a simple value alteration as follows. Take a random value from the individual X and consider that the value at this position is x_i with genome $genome(x_i) = b_j b_{j-1} \dots b_1$. Choose a random bit position k, such that $1 \leqslant k \leqslant j$. Then a new individual is obtained just by replacing the bit b_k by its negation counterpart, $b'_k = not(b_k)$.

Although the crossover definition says that it uses two different individuals to cross them and obtain two new offspring, in our case, we use the whole population as a single solution and we adapted this operator to our case.

Crossover of the individual X is performed by swapping two ranges of values inside the individual as follows. Take two value positions $s,\ r$ at random, and consider that the two values at this position are $x_s \in X$ and $x_r \in X$. Generate a random number m to indicate the ranges length which must be in the range $[0, min(length(X) - s, length(X) - r, |s - r|)]$, where $length(X)$ is the total number of values inside the individual X, and $\|$ is the absolute value operator. Then the ranges $[x_s, x_{s+m}]$ and $[x_r, x_{r+m}]$ are swapped obtaining a new individual. For example, having $s < r$ and $X = \{x_1, \dots, x_n\}$ the new individual will be $X' = \{x_1, \dots, x_r, \dots, x_{r+m}, \dots, x_s, \dots, x_{s+m}, \dots, x_n\}$.

$$\begin{matrix} 0.185 & 0.283 & 0.532 \\ 0.609 & 0.089 & 0.302 \\ 0.002 & 0.277 & 0.721 \end{matrix}$$

PRAM matrix.

$$\begin{matrix} 185 & 283 & 532 \\ 609 & 89 & 302 \\ 2 & 277 & 721 \end{matrix}$$

PRAM matrix in integers mode.

$$\begin{matrix} 0010111001 & 0100011011 & 1000010100 \\ 1001100001 & 0001011001 & 0100101110 \\ 0000000010 & 0100010101 & 1011010001 \end{matrix}$$

Binary matrix.

$$\begin{matrix} 0011100101 & 0110010110 & 1100011110 \\ 1101010001 & 0001110101 & 0110111001 \\ 0000000011 & 0110011111 & 1110111001 \end{matrix}$$

Gray-coded genome matrix.

Fig. 1. Example of genotype encoding.

Table 1
Initial PRAM matrix using Eq. (1) with $p = 0.5$ for the attribute DEGREE in the U.S. Housing Survey dataset. The bold values represent the most probable category where each one will be changed to when using the PRAM matrix to protect the data.

0.500	0.067	0.060	0.065	0.067	0.076	0.080	0.083
0.073	**0.500**	0.058	0.064	0.066	0.075	0.080	0.083
0.072	0.064	**0.500**	0.062	0.064	0.074	0.080	0.083
0.073	0.065	0.057	**0.500**	0.066	0.075	0.080	0.083
0.073	0.066	0.058	0.064	**0.500**	0.075	0.080	0.083
0.074	0.068	0.061	0.066	0.068	**0.500**	0.080	0.083
0.075	0.068	0.062	0.067	0.069	0.076	**0.500**	0.083
0.075	0.069	0.062	0.067	0.069	0.077	0.081	**0.500**

Table 2
Frequencies of the DEGREE attribute in the U.S. Housing Survey dataset.

Categories	'1'	'2'	'3'	'4'	'5'	'6'	'9'	'–'
Frequency	98	173	251	195	170	80	33	0

Table 3
Initial and final Scores for the protection of the three attributes DEGREE, BUILT, GRADE1 in U.S. Housing dataset at the same time.

	IL	DR	Score
Initial	63.14	31.08	47.11
Final	53.77	8.61	31.20

3.3. Fitness function

The evolutionary algorithm needs a mechanism to check the quality of the individuals in the population (in both the initial ones and the new offspring in each generation). The fitness function is this mechanism.

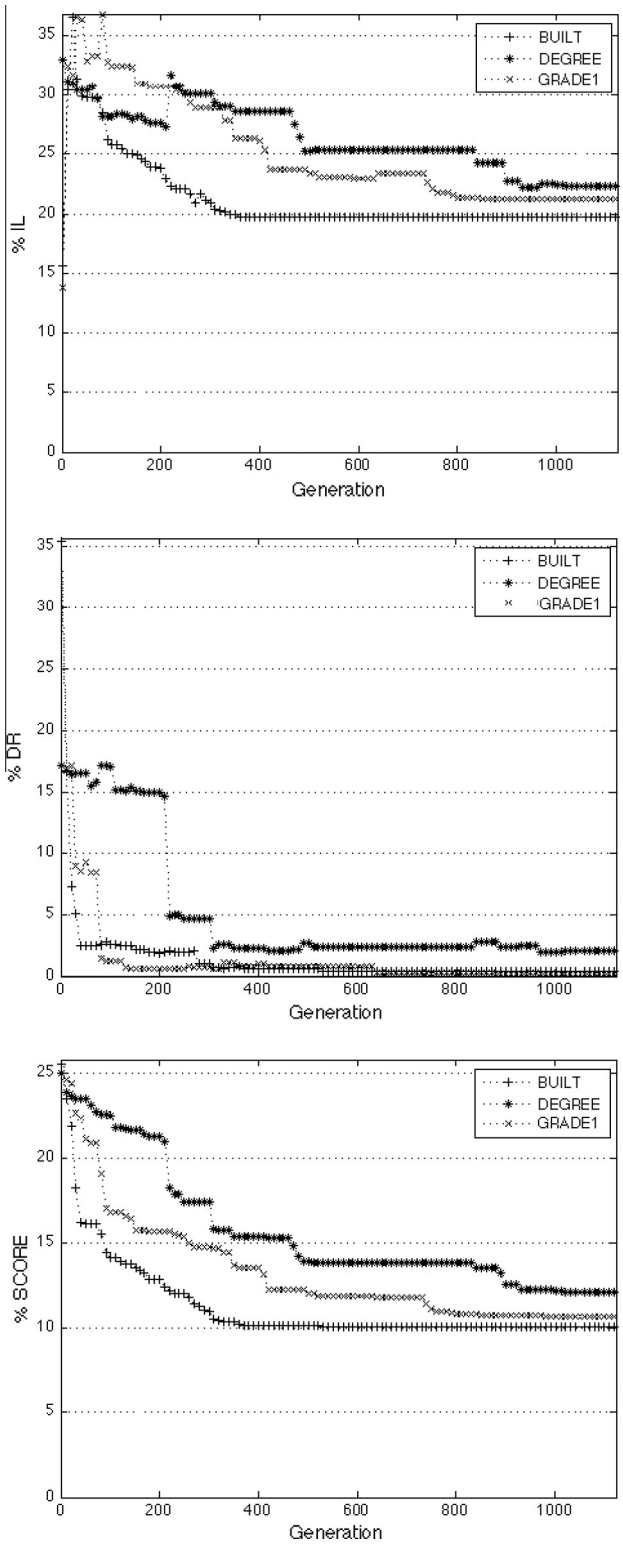

Fig. 2. Evolution of the measures for the individual protection of the three attributes BUILT, DEGREE and GRADE1 in U.S. Housing dataset.

In our case, the evaluation of PRAM matrices need several steps before checking their protection quality. First of all, these PRAM matrices values are in Gray code representation so it is needed to restore them to floating point values. Then, it is not possible to check the quality of the matrices just by taking a look at them so, as a second step, we use these matrices to perform the multivariate PRAM protection on the original data obtaining a certain protected dataset. After this second step we are finally able to check the protection quality using the two measures described in Section 2.2: Information Loss and Disclosure Risk.

As there are two measures to be taken into account, it can be considered as a multi-objective optimization problem. To solve this we chose a multi-objective optimization method called Objective Weighting which allows us to combine both measures applying an individual weight to each one. We wanted to give the same importance to both Disclosure Risk (DR) and Information Loss (IL) measures, so both have $\frac{1}{2}$ as a weight value.

If F is the original file and $PRAM_{multivariate}(F, \{X_1, \ldots, X_n\})$ is the function that performs multivariate PRAM protection in F with the set of PRAM matrices $\{X'_1, \ldots, X'_n\}$, then, the score of the set of matrices $\{X_1, \ldots, X_n\}$ is computed as follows

$$\{X'_1, \ldots, X'_n\} = restore(\{X_1, \ldots, X_n\}) \tag{4}$$

$$F' = PRAM_{multivariate}(F, \{X'_1, \ldots, X'_n\}) \tag{5}$$

$$Score(\{X'_1, \ldots, X'_n\}) = \frac{DR(F') + IL(F')}{2} \tag{6}$$

where $DR()$ is the Disclosure Risk evaluation function and $IL()$ is the Information Loss evaluation function.

Because the PRAM method takes random decisions in the protection step, the method can generate different protected files for the same Markov matrix, and they will also have different scores. In order to have more robust results, we compute 5 protected files for each candidate to be evaluated (i.e. each Markov matrix) and the average of their scores is taken as the candidate's final score. It should be noticed that the number of executions to perform is not fixed and it can be changed by the user. We used 5 executions because with it we obtained enough robust results without penalizing too much the execution time. More formally:

$$FinalScore(\{X_1, \ldots, X_n\}) = \frac{\sum_{i=1}^{5} Score(\{X'_1, \ldots, X'_n\})}{5} \tag{7}$$

4. Experimental results

In order to illustrate and empirically evaluate our proposed method we used three different datasets to perform some experiments. The first dataset is a U.S. Housing Survey of 1993 from the U.S. Census Bureau [20] with 1000 records and

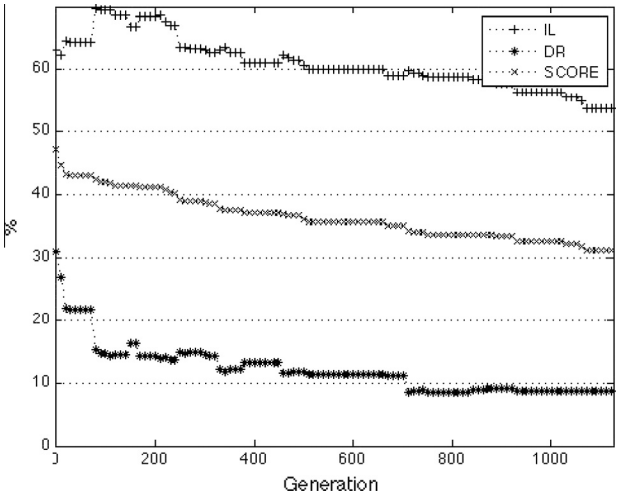

Fig. 3. Results for the protection of all three attributes at the same time in the U.S. Housing dataset.

Table 4
Initial and final Scores for the protection of the three attributes EXISTACC, PRESEMPLOY, and SAVINGS in the German dataset at the same time.

	IL	DR	Score
Initial	80.36	25.63	52.99
Final	38.39	27.30	32.84

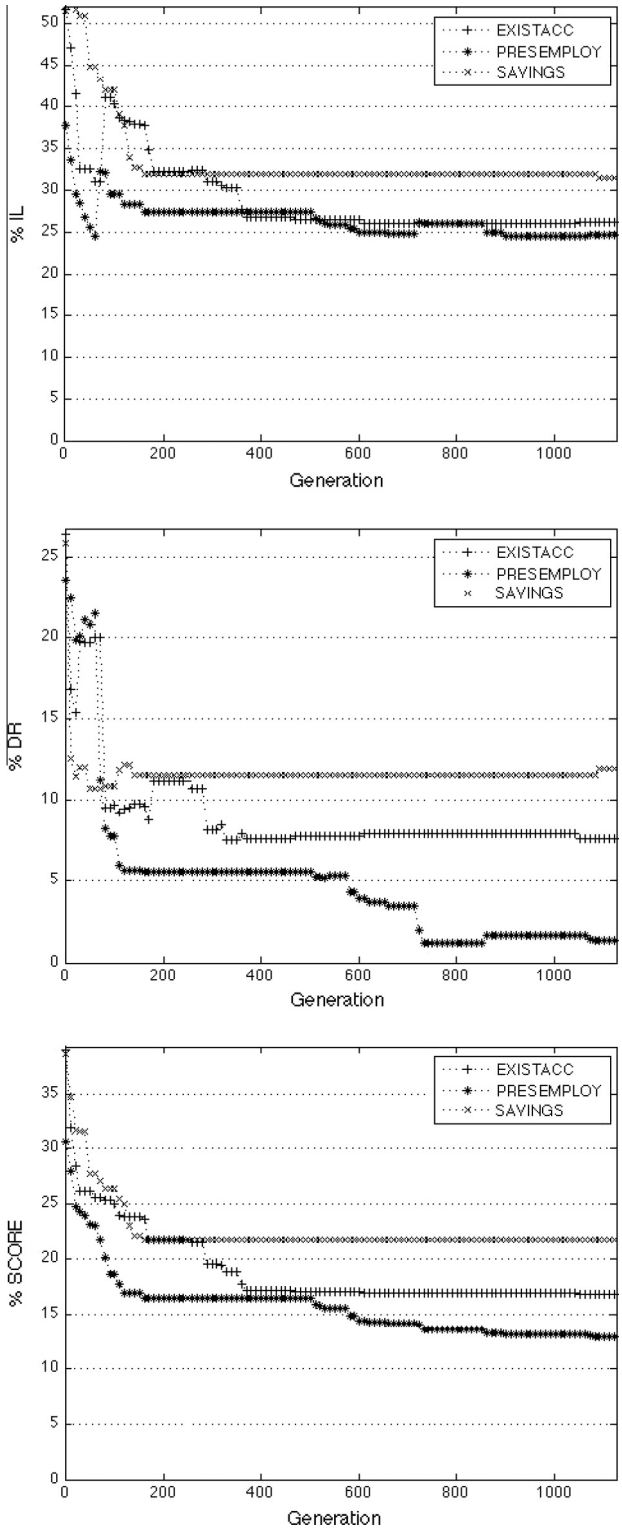

Fig. 4. Evolution of the measures for individual protection of the three attributes EXISTACC, PRESEMPLOY, and SAVINGS in the German dataset.

11 categorical attributes. The second dataset contains information about credit risk of German people [10], and consists of 1000 records and 20 attributes. Finally the third dataset contains information about Solar Flares [10], and consists of 1389 records and 10 attributes.

In these experiments we chose $p = 0.5$ as a parameter value to create the initial Markov matrices. This value represents the quantity of original values that are wanted to be kept after perturbation (in this case we want to keep only 50% of the original values). Its value is up to the data user and, in this work, we used this value to demonstrate the ability to find good matrices from a bad one. As an example, Table 1 shows the initial Markov matrix for the DEGREE attribute corresponding to the U.S. Housing Survey data set with 8 categories, and Table 2 shows the frequencies of the categories corresponding to this attribute in the original data set. Note that the max values in each row are highlighted.

It is easy to see that the higher off-diagonal values corresponds to the attributes that have less frequency inside the original data set. This effect makes that, after the protection process, the frequencies of all categories are more balanced in order to increase the uncertainty inside the data set. Then, the problem here is to obtain a good PRAM matrix in order to achieve a better protection minimizing the Information Loss and the Disclosure Risk. (see Table 3).

Our proposed method *a priori* applies to any particular PRAM matrix, but of course it is generally better to start from a *good* matrix.

Each experiment is divided in two phases. In the first phase, some attributes are going to be protected independently checking the performance of our method when only one attribute is protected, while in the second phase all the previous attributes are going to be protected together at the same time checking the performance of our method when multi-attribute protection is performed.

In our first experiment we used the U.S. Housing Survey dataset considering 3 attributes to protect and their respective PRAM matrices. The first attribute is named DEGREE and it has 8 different ordinal categories, the second one is the BUILT attribute with 25 different ordinal categories, and finally, the third attribute is named GRADE1 with 21 different ordinal categories.

Starting with the first experiment, Fig. 2 shows the results for the measures evolution for the individual protections of the three attributes (BUILT, DEGREE, and GRADE1) during all the 1125 generations.

As it can be seen, Information Loss and Disclosure Risk are being adjusted in order to reduce the Score value. It does not mean that all measures are decreasing all the time (e.g. there is an increment of the Information Loss at generation 200). Score is the only measure that is strictly decreasing its value and never increases, so this implies that the result at any generation will be at least as good as the previous one.

Looking at the decrement of the Score we can see that, during the optimization approach that we propose, the measure has been reduced quite significantly in all the cases. It means that the new PRAM matrix performs a much better protection than the original one.

Once we know that our approach is working quite good when protecting isolated attributes, a multi-attribute protection can also be performed.

In Fig. 3 we can find the evolution of the Information Loss, Disclosure Risk and Score for this multi-attribute protection. During the evolutionary process it can be seen that there is a progressive decrement until around generation 1100 of the Score value during all the process. Moreover, Disclosure Risk has suffered a big decrement, so the combination of this decrement with the little reduction of Information Loss has forced the Score value to be reduced. In this case, Score measure started with a value of 47.11% and, after the evolutionary process, it ended with a value of 31.20%. This represents a decrement of a 33.77%. Table 4 shows the initial and final results of the three measures.

In the second experiment we used the German dataset considering 3 attributes to protect and their respective PRAM matrices. This dataset has attributes with less categories than the dataset used in the first experiment. So, with this second experiment we are going to prove that our approach works well also with this kind of attributes. The first attribute is named

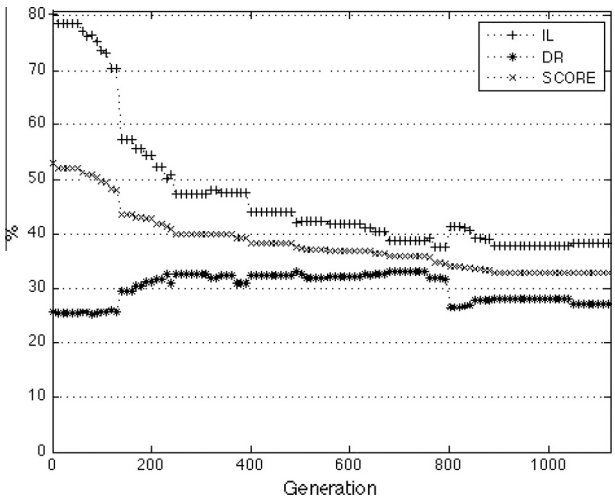

Fig. 5. Results for the protection of all three attributes EXISTACC, PRESEMPLOY, and SAVINGS at the same time in the German dataset.

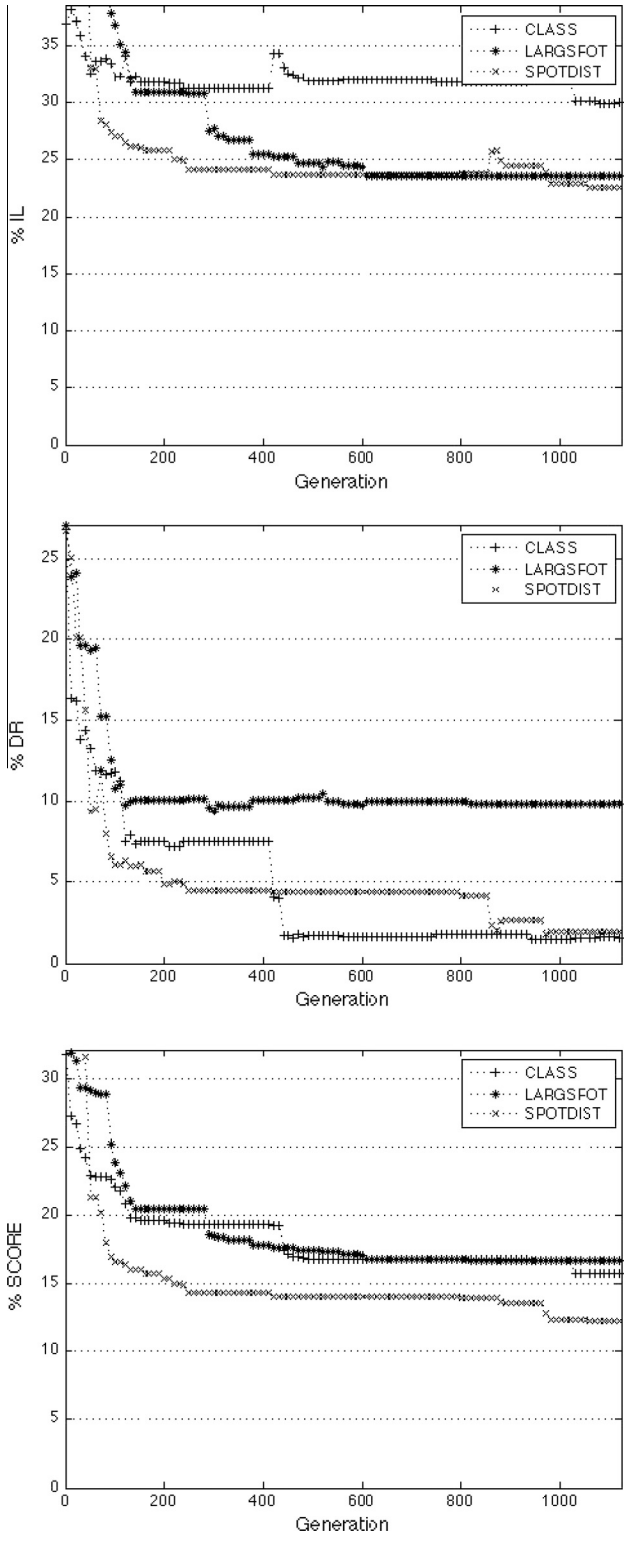

Fig. 6. Evolution of the measures for the individual protection of the three attributes CLASS, LARGSPOT, and SPOTDIST in Solar Flare dataset.

EXISTACC with 5 different ordinal categories, the second one is PRESEMPLOY with 6 different ordinal categories, and finally, the third attribute is named SAVINGS with 6 different ordinal categories.

The evolution of its Information Loss, Disclosure Risk and Score measures of all the individual protections for the attributes are shown in Fig. 4. It can be seen that Disclosure Risk measures have been reduced so much (specially in the case of PRESEMPLOY) with a big decrement in the first generations. The decrement of the Information Loss measures is smaller but it also exhibits the main decrement in the first generations like in the case of Disclosure Risk. Moreover, the measures of Information Loss and Disclosure Risk are being adjusted in an irregular way (there are some increments and decrements of their values while Score is being reduced).

It has been proved that our approach is working well in protecting one ordinal attribute with only few categories. Now in the last part of this second experiment all three attributes are going to be protected together in order to test our approach in a multi-attribute protection for this kind of attributes.

Fig. 5 shows the evolution of the Information Loss, Disclosure Risk and Score for this multi-attribute protection. During the evolutionary process it can be seen that there is a quite progressive decrement of the Score value during all the process. Moreover the Disclosure Risk has increased instead of decreased, but its final value is quite close to the initial one while the Information Loss has suffered a big decrement, so the combination of the two measures forced to reduce the Score value. In this case, the Score measure started with a value of 52.99% and after the evolutionary process it ended with a value of 32.84%. This represents a decrement of a 38.03%. Table 4 shows the initial and final results of the three measures.

Finally in our third experiment we used the Solar Flare dataset considering 3 attributes to protect with their respective PRAM matrices. The difference of this dataset is that we are going to protect nominal attributes instead of ordinal attributes like in the previous experiments in order to prove that our approach works also well with this kind of attributes. The first attribute is named CLASS with 8 different nominal categories, the second one is LARGSPOT with 7 different nominal categories, and finally, the third attribute is named SPOTDIST with 5 different nominal categories.

First we protected each attribute independently obtaining the results shown in Fig. 6. In this case there is a slow decrement for the Information Loss measures while Disclosure Risk is having a very big decrement in the first generations. Looking at the evolution of the Score measure, we see that it has a quite regular and important decrement during all the process.

Next step is to protect all three attributes together in order to test our approach in a multi-attribute protection for this kind of attributes.

Fig. 7 shows the behavior of Information Loss, Disclosure Risk and Score for this multi-attribute protection in this Solar Flare dataset. In this figure it can be seen that all three measures have a fast stabilization around generation 600. Disclosure Risk has increased a little but Information Loss has been suffered a big decrement which causes an important reduction to the values of the Score measure. In Table 5 we see that the Score measure has started the evolutionary process with a value of 53.83% and after the evolutionary process it ended with a value of 33.16%. This represents a decrement of a 38.40%.

There is also another interesting point to discuss. That is the changes that have been performed between the original matrices and the final ones. Table 6 shows the final matrix of the DEGREE attribute (initial matrix is shown in Table 1). Note that, as in the initial matrix, the maximum values in each row are highlighted.

It is easy to see that after the evolutionary process the new matrix has a lot of changes, but the most remarkable one is that, in general, all the highest row values (which are the ones corresponding to the category with highest probability to substitute the original one) are outside of the diagonal, whereas in the initial matrices, largest values were all in the diagonal.

Nevertheless, it is very difficult to find a model of matrix or a general pattern that is the best for all problems. That is so because the best matrices obtained in the different evolutions have different structures and different probability distributions over the rows. To find such a model without the effective model of our approach using genetic algorithms is an open problem.

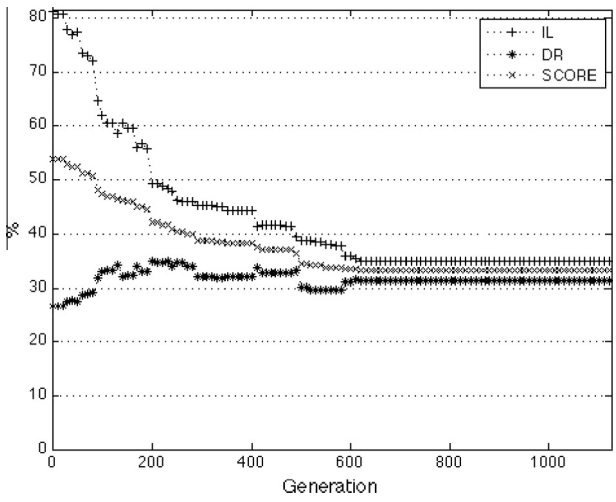

Fig. 7. Results for the protection of all three attributes CLASS, LARGSPOT, and SPOTDIST at the same time in Solar Flare dataset.

Table 5
Initial and final Scores for the protection of the three attributes CLASS, LARGSPOT, and SPOTDIST at the same time in the Solar Flare dataset.

	IL	DR	Score
Initial	81.18	26.49	53.83
Final	34.91	31.41	33.16

Table 6
Final PRAM matrix with $p = 0.5$ for DEGREE attribute corresponding to U.S. Housing Survey dataset. The bold values represent the most probable category where each one will be changed to when using the PRAM matrix to protect the data.

0.011	0.009	0.011	0.009	**0.926**	0.009	0.011	0.014
0.000	0.004	**0.971**	0.005	0.005	0.005	0.004	0.004
0.027	**0.625**	0.009	0.027	0.205	0.036	0.036	0.036
0.049	0.037	**0.432**	0.074	0.062	0.259	0.049	0.037
0.032	0.005	0.006	**0.928**	0.006	0.008	0.006	0.008
0.892	0.008	0.033	0.008	0.011	0.017	0.017	0.014
0.003	0.030	0.004	0.004	0.466	**0.487**	0.003	0.003
0.430	0.416	0.005	0.006	0.019	0.006	0.006	0.111

Table 7
Initial and final categories frequencies in DEGREE attribute corresponding to U.S. Housing Survey dataset.

Categories	'1'	'2'	'3'	'4'	'5'	'6'	'9'	'–'
Initial freq.	98	173	251	195	170	80	33	0
Final freq.	96	171	250	181	181	81	17	23

Another fact that can be seen is that, in the final protected dataset, the frequencies of the categories are more balanced than in the original dataset, and the matrix also introduces more uncertainty to the protected dataset (i.e. the final distribution entropy is bigger than the one of the initial distribution). Table 7 shows the differences between initial and final frequencies inside the dataset after the protection.

5. Conclusions and future work

In this paper we have proposed an optimization method for PRAM using an evolutionary algorithm in order to seek new enhanced PRAM matrices to perform better protections for categorical attributes protecting either one or multiple attributes at the same time. It has been tested with three dataset with different properties: the U.S. Housing dataset with ordinal attributes and large number of categories per attribute, the German dataset with ordinal attributes and only few number of categories per attribute, and the Solar Flare dataset with nominal attributes. In all the cases we obtained very good results.

For the first dataset we obtained an improvement for a multi-attribute protection of a 33.77%, for the second dataset the improvement was of a 38.03%, and finally for the third dataset we obtained a 38.40% of improvement. This result represents the effectiveness of our approach.

Evolutionary algorithms have the advantage that they can be adapted very easily to other fitness functions. This is important for our approach because, as it is based on an evolutionary algorithm, it can be adapted to other possible future new measures of Information Loss and Disclosure Risk just by providing a different fitness function. This decoupling of the algorithm from protection measures is an interesting property that might deserve future research.

As a disadvantage there is the computational cost of the current Information Loss and Disclosure Risk measures that guide the fitness evaluation at each generation, but it can be compensated by the property of evolutionary algorithms where the implementation may be parallelized in order to reduce performance costs. This can be taken into account for other future optimizations exploration.

Another line for future work was outlined in Section 4. That is find general models of matrices from the ones obtained by our approach that lead to good scores. We have seen that the genetic algorithm is able to improve in a significant way the results of the usual Markov matrices used by PRAM. Nevertheless, finding an analytical definition of these effective matrices would reduce the burden of using genetic algorithms.

Acknowledgments

This work has been done under the PhD in Computer Science program of the Universitat Autònoma de Barcelona (UAB). It is also partially supported by the Spanish MEC ARES-CONSOLIDER INGENIO 2010 CSD2007-00004, and COPRIVACY TIN2011-27076-C03-03. The research leading to these results has also received funding from the European Union's Seventh Framework Programme (FP7/2007-2013) under Grant Agreement Num. 262608.

References

[1] R. Agrawal, R. Srikant, Privacy preserving data mining, in: Proceedings of the ACM SIGMOD Conference on Management of Data, 2000, pp. 439–450.
[2] T. Back, D.B. Fogel, Z. Michalewicz (Eds.), Evolutionary Computation, Advanced Algorithms and Operations, vol. 2, Institute of Physics Publishing, 2000.
[3] R.A. Caruana, J.D. Schaffer, Representation and hidden bias: Gray vs. binary coding for genetic algorithms, in: Proceedings of the 5th International Conference on Machine Learning, Morgan Kaufmann, Los Altos, CA, 1988, pp. 153–161.
[4] E. Cator, A. Hensbergen, Y. Rozenholc, Statistical Disclosure Control using PRAM, DUP Science, Delft, 2005. pp. 23–33.
[5] P. De Wolf, V. Gelder, An Empirical Evaluation of PRAM. Discussion Paper No. 04012, Statistics Netherlands, Voorburg/Heerlen, 2004.
[6] J. Domingo-Ferrer, V. Torra, Disclosure control methods and information loss for microdata, in: P. Doyle, J.I. Lane, J.J.M. Theuwes, L. Vatz (Eds.), Confidentiality, Disclosure and Data Access: Theory and Practical Applications for Statistical Agencies, Elsevier, 2001, pp. 91–110 (Chapter 5).
[7] J. Domingo-Ferrer, V. Torra, A quantitative comparison of disclosure control methods for microdata, in: P. Doyle, J.I. Lane, J.J.M. Theuwes, L. Zayatz (Eds.), Confidentiality, Disclosure, and Data Access: Theory and Practical Applications for Statistical Agencies, Elsevier, 2001, pp. 111–133.
[8] J. Domingo-Ferrer, V. Torra, Distance-based and probabilistic record linkage for re-identification of records with categorical variables, in Butlletí de l'ACIA, vol. 28, Associació Catalana d'Intel.ligència Artificial, 2002, pp. 243–250.
[9] S. Fienberg, Conflict between the needs for access to statistical information and demands for confidentiality, J. Off. Stat. 10 (1994) 115–132.
[10] A. Frank, A. Asuncion, UCI Machine Learning Repository. University of California, Irvine, School of Information and Computer Sciences, 2010. <http://archive.ics.uci.edu/ml>.
[11] J. Gouweleeuw, P. Kooiman, L. Willenborg, P. de Wolf, Post randomization for statistical disclosure control: theory and implementation, J. Off. Stat. 14 (1998) 463–478.
[12] D. Greiner, G. Winter, J.M. Emperador, B. Galván, Gray coding in evolutionary multicriteria optimization: application in frame structural optimum design, in: Proceedings of the Third international Conference on Evolutionary Multi-Criterion Optimization, Springer-Verlag, Berlin, Heidelberg, 2005, pp. 576–591.
[13] J.H. Holland, Adaptation in Natural and Artificial Systems, University of Michigan Press, Ann Arbor, MI, USA, 1975.
[14] A. Hundepool, J. Domingo-Ferrer, L. Franconi, S. Giessing, E.S. Nordholt, K. Spicer, P.P. de Wolf, Statistical Disclosure Control, Wiley, 2012.
[15] A. Hundepool, L. Willenborg, Argus: software from the SDC project, in: Proceedings of Joint UNECE–Eurostat Work Session on Statistical Data Confidentiality, Luxembourg: UNECE–Eurostat, 1998, pp. 87–98.
[16] J. Marés, V. Torra, PRAM optimization using an evolutionary algorithm, in: J. Domingo-Ferrer, E. Magkos (Eds.), Privacy in Statistical Databases, Springer, 2010, pp. 97–106.
[17] Z. Michalewicz, D.B. Fogel, How to Solve It: Modern Heuristics, second ed., Springer, 2004.
[18] P. Samarati, Protecting respondents' identities in microdata release, IEEE Trans. Knowl. Data Eng. 13 (2001) 1010–1027.
[19] V. Torra, Microaggregation for categorical variables: a median based approach, in: J. Domingo-Ferrer, V. Torra (Eds.), Privacy in Statistical Databases, Springer, 2004, pp. 162–174.
[20] U.S. Census Bureau, U.S. Housing Survey of 1993, 1993. <http://quickfacts.census.gov>.
[21] L. Willenborg, T. de Waal, Statistical Disclosure Control in Practice, Springer, 1996.
[22] P.P.D. Wolf, J.M. Gouweleeuw, P. Kooiman, L. Willenborg, Reflections on PRAM, Statistics Netherlands, Department of Statistical Methods, 1999.