2007 International Conference on Intelligent Pervasive Computing

Increasing Polynomial Regression Complexity for Data Anonymization

Jordi Nin[1] Jordi Pont-Tuset[2] Pau Medrano-Gracia[2]

Josep L. Larriba-Pey[2] Victor Muntés-Mulero[2]

[1]IIIA, Artificial Intelligence Research Institute
CSIC, Spanish National Research Council
Universitat Autònoma de Barcelona
jnin@iiia.csic.es

[2]DAMA-UPC
Universitat Politècnica de Catalunya
Campus Nord UPC, C/Jordi Girona 1-3
08034 Barcelona, (Catalonia, Spain)
{jpont,pmedrano,larri,vmuntes}@ac.upc.edu

Abstract

Pervasive computing and the increasing networking needs usually demand from publishing data without revealing sensible information. Among several data protection methods proposed in the literature, those based on linear regression are widely used for numerical data. However, no attempts have been made to study the effect of using more complex polynomial regression methods. In this paper, we present PoROP-k*, a family of anonymizing methods able to protect a data set using polynomial regressions. We show that* PoROP-k *not only reduces the loss of information, but it also obtains a better level of protection compared to previous proposals based on linear regressions.*

I. Introduction

Pervasive computing together with the use of the Internet is becoming very popular in a lot of business and research areas. In this situation, publicly accessible networks are populated with large amounts of sensible information that needs to be protected. Thus, privacy becomes a priority. Surveys show that most of web users are unwilling to provide confidential data to a web site unless privacy protection measures are provided [2].

A wide range of anonymizing methods have been proposed in the literature. The goal of these methods is to ensure an acceptable level of protection of the confidential data preserving their statistical utility. Good surveys about protection methods may be found in [1], [8].

In [8], the anonymizing methods are classified into two different categories depending on how they use the original values: *synthetic data generators* and *perturbative protection methods*. Synthetic data generators exclusively use the original data to build a model that is later used to build a new data set. Perturbative protection methods are based on the addition of noise into the original data set in order to make it difficult for an intruder to discover the original values.

Linear regression models are commonly used to anonymize data. Two examples of this kind are the *Information Preserving Statistical Obfuscation* (IPSO) [4], a synthetic data generator, and the methods included in the LiROP-k family [10]. The latter include both perturbative protection and synthetic data generation, and were developed to solve some drawbacks of IPSO. However, to our knowledge, more complex regression methods have not been presented in the literature.

In this paper, we study a new family of methods called *PoROP-k*, that make it possible to protect confidential data by using more complex regression models. We show in our experiments that increasing the complexity of the regression model, *PoROP-k* methods outperform LiROP-k methods (which are a particular case of the family of methods included in *PoROP-k*), when the *score*, a standard measure to compare protection methods defined in [7], is used to compare both methods.

The structure of the paper is as follows. Section II depicts the protection scenario assumed in this work. In Section III, we present our protection method using polynomial regression. Then, Section IV describes the experiments done. Finally, the paper draws some conclusions and a description of future work.

II. Privacy Protection Scenario

Before presenting our proposal, we first present the protection scenario assumed in this work.

The main objective of a protection method is to anonymize a data set. A data set can be viewed as a file containing a number of records, where each record contains a set of attributes of an individual. The attributes in the original data set can be classified into two different categories, depending on their capability to identify unique individuals, as follows:

- **Identifiers.** The identifier attributes are used to identify the individual unambiguously. A typical example of identifier is the passport number.

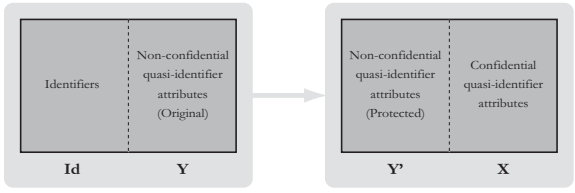


Fig. 1. Re-identification scenario.

0-7695-3006-0/07 $25.00 © 2007 IEEE
DOI 10.1109/IPC.2007.103

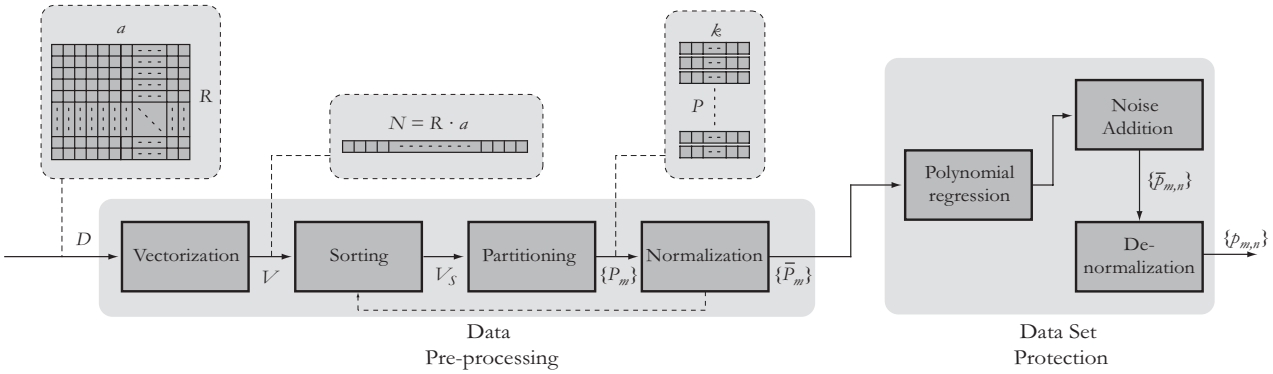


Fig. 2. *PoROP-k* protection schema.

- **Quasi-identifiers.** A quasi-identifier attribute is an attribute that is not able to identify a single individual when it is used alone. However, when it is combined with other quasi-identifier attributes, they can uniquely identify an individual. Among the quasi-identifier attributes, we distinguish between confidential and non-confidential, depending on whether they contain confidential information. An example of non-confidential quasi-identifier attribute would be the postal code, while a confidential quasi-identifier might be the salary.

When a data set is protected, identifiers are removed or encrypted to prevent an intruder to re-identify individuals easily. Typically, the remaining attributes are released, some of them protected. In this paper, we assume that non-confidential attributes are protected, while confidential attributes are not. This allows third parties to have precise information on confidential data without revealing to whom that confidential data belongs to.

In this scenario, as shown in Figure 1, an intruder might try to re-identify individuals by obtaining the non-confidential quasi-identifier data (Y) together with identifiers (*Id*) from other data sources. Applying record linkage between the protected attributes (Y') and the same attributes obtained from other data sources (Y), the intruder might be able to re-identify a percentage of the protected individuals together with their confidential data (X). This is what protection methods try to prevent.

III. Method description

Analogously to *LiROP-k* methods, Polynomial Regression on Ordered Partitions (*PoROP-k*) methods pre-process the original data using a sequence of basics steps, namely (i) vectorization, (ii) sorting, (iii) partitioning and (iv) normalization. The whole anonymizing process is depicted in Figure 2. There are several aspects that motivate these steps:

- **Vectorization.** The main idea of this first step is to gather all the values in the data set in a single vector, independently of the attribute they belong to. Consequently, we are ignoring the attribute semantics and, therefore, all possible relations, like covariance or correlations among the attributes in the data set.
- **Sorting.** The second step is to sort all the vectorized values. This step is necessary in order to fit the data into a model easily, sorting the values in non-decreasing order. Note that sorting the values is a way of adding noise itself. Note also that, performing this step, we are not losing information as we will see later in the results.
- **Partitioning.** Even taking into account that data is sorted, using a unique model to fit all the data is unfeasible because the error of the model might be very large. In order to improve the accuracy, the sorted vectorized data is split into several k-partitions, where k is the number of values per partition. When all the pre-process steps are finished, a different model regression will be used to fit the data of each partition. Therefore, modifying the value of k, *PoROP-k* methods allow us to tune the accuracy of the regression model by changing the size of the partition being fitted. Note that if the data set was not sorted, k would not have this property.
- **Normalization.** Since the range of the values in the different attributes could differ significantly among them, it might happen that the sorting step does not merge all the attributes appropriately. For this reason, it is necessary to normalize the data. There are many ways to normalize a data set. A possible solution would be to normalize each attribute independently before the application of the vectorization step. However, this normalization method could present problems with skewed attributes and, therefore, the attributes could not be merged in the sorting step. For this reason, we propose to normalize the data stored in each partition independently. This way, similar values are put in the same partition and, therefore, the chances to avoid the effect of skewness in the data are higher.

Note that, once data is normalized, vectorization, sorting and partitioning steps have to be repeated a second time. This step allows mixing values of different attributes independently from their domain.

Formally speaking, let $\mathcal{D}$ be the original data set to be protected. We denote by R the number of records in $\mathcal{D}$. Each

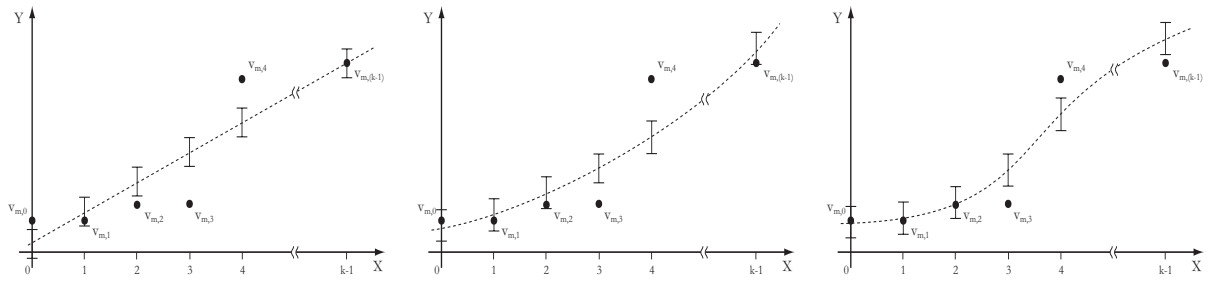


Fig. 3. Example of different regression curves on a set of points from a partition and the more probable intervals for the protected values when noise is added independently.

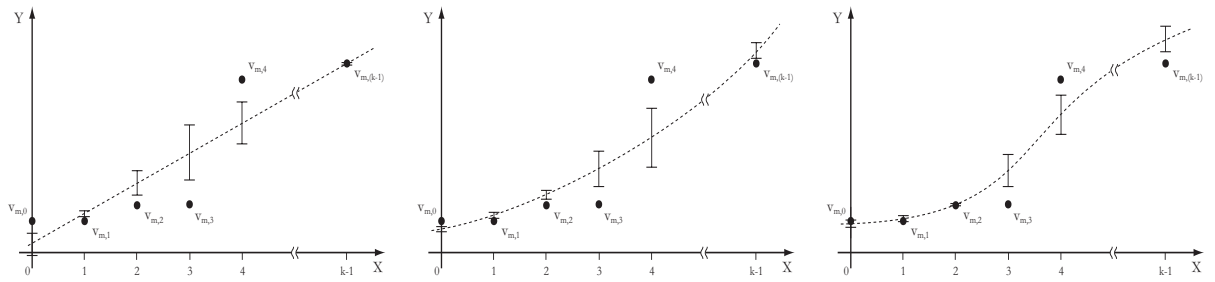


Fig. 4. Example of different regression curves on a set of points from a partition and the more probable interval for the protected values when noise is added taking into account the original value.

record consists of a numerical attributes or fields. We assume that none of the registers contain blanks. We denote by N the total number of values in $\mathcal{D}$. As a consequence, $N = R \cdot a$.

Let V be a vector of size N containing the data values. First, V is sorted increasingly. Let us denote by V_s the ordered vector of size N containing the sorted data and v_i the ith element of vector V_s, where $0 \leq i < N$.

Next, V_s is divided into smaller sub-vectors or partitions. Then, each sub-vector is normalized into the $[0, 1]$ interval and they are all sorted and partitioned again. As we said before, we formally define k, where $1 < k \leq N$, as the number of values per partition. Note that, if k is not a divisor of N the last partition will contain a smaller number of values. Let P be the number of k-partitions. We call r the number of values in the last partition where $0 \leq r < k$. Therefore, $N = kP + r$. If $r > 0$, we have $P + 1$ partitions. We denote by P_m the mth partition.

Let $v_{m,n}$ be defined as the nth element of P_m:

$$\begin{cases} v_{m,n} := v_{mk+n} \; n = 0 \ldots k-1 \; m = 0 \ldots P-1 \\ v_{P,n} := v_{Pk+n} \; n = 0 \ldots r-1 \end{cases}$$

For each P_m, a regression model is computed over the following $(x,\ y)$ points:

$$(0\,,\, v_{m,0}) \quad (1\,,\, v_{m,1}) \quad \cdots \quad (k-1\,,\, v_{m,(k-1)})$$

When $r > 0$, the size of the last partition (P_P) is $r < k$. In this case, the regression model of this partition is computed differently: the nearest last k points of the data set are used to compute the regression model, but only the r points held by P_P are actually protected. This guarantees that each regression model is computed using the same number of points, so the level of accuracy is homogeneous. Therefore, in this case, the fitting for the last partition is computed over the following $(x,\ y)$ points:

$$(0\,,\, v_{m,N-k})\ (1\,,\, v_{m,N-k+1}) \cdots (k-1\,,\, v_{m,N-1})$$

Finally, when the regression model is computed, *PoROP-k* methods add Gaussian noise to the polynomial regression to partially change the order of the points. With the addition of noise, it will be more difficult for an intruder to reveal the original data even knowing the values of some attributes.

Similarly to LiROP-k methods, *PoROP-k* methods may be considered both a protection method and a synthetic data generator depending on the way used to add noise. If the Gaussian noise is computed independently of the original value to protect, *PoROP-k* methods can be considered synthetic data generators. We call this set of methods *PoROP$_s$-k*. An example of three different regression curves computed on the same data set using different polynomial complexity is described in Figure 3. On the other hand, if the noise addition is dependent on the point to be protected, *PoROP-k* methods must be considered perturbative. In this latter case, we call this methods set *PoROP$_p$-k*. An example of these is presented in Figure 4. More details about noise addition methods based on linear regression models can be found in [10].

$$\gamma_m = 30\,\frac{\sum_{n=0}^{k-1} v_{m,n}}{k\,(k^2+3\,k+2)} - 180\,\frac{\sum_{n=0}^{k-1} n\,v_{m,n}}{k\,(k^3+k^2-4\,k-4)} + 180\,\frac{\sum_{n=0}^{k-1} n^2 v_{m,n}}{k\,(k^4-5\,k^2+4)}$$

$$\beta_m = -18\,\frac{(2\,k-1)\sum_{n=0}^{k-1} v_{m,n}}{k\,(k^2+3\,k+2)} + 12\,\frac{\left(16\,k^2-30\,k+11\right)\sum_{n=0}^{k-1} n\,v_{m,n}}{k\,(k^4-5\,k^2+4)} - 180\,\frac{\sum_{n=0}^{k-1} n^2 v_{m,n}}{k\,(k^3+k^2-4\,k-4)}$$

$$\alpha_m = 3\,\frac{\left(3\,k^2-3\,k+2\right)\sum_{n=0}^{k-1} v_{m,n}}{k\,(k^2+3\,k+2)} - 18\,\frac{(2\,k-1)\sum_{n=0}^{k-1} n\,v_{m,n}}{k\,(k^2+3\,k+2)} + 30\,\frac{\sum_{n=0}^{k-1} n^2 v_{m,n}}{k\,(k^2+3\,k+2)}$$

Fig. 5. Equations to model a data set using quadratic regression.

Following, we present the formulae to compute *PoROP-k* methods using linear and quadratic regressions. Although formulas for cubic regressions are used later in the experiments, the details about them are omitted in this paper for the sake of simplicity.

A. PoROP-k using linear regression

Assuming that the resulting linear regression which models the data is $l_{m,n} = \beta_m n + \alpha_m$ (where $n = 0 \ldots k-1$), then the expressions used to compute β_m and α_m are as follows:

$$\beta_m = \frac{2}{k(k+1)}\left[-3\sum_{n=0}^{k-1} v_{m,n} + \frac{6}{k-1}\sum_{n=1}^{k-1} n\,v_{m,n}\right]$$

$$\alpha_m = \frac{2}{k(k+1)}\left[(2k-1)\sum_{n=0}^{k-1} v_{m,n} - 3\sum_{n=1}^{k-1} n\,v_{m,n}\right]$$

These results can be derived from the normal equations as presented in [5].

Note that, as mentioned before, when linear regression is used to model the data in each partition, *PoROP-k* methods can be reduced to *LiROP-k* methods, since this last subset is a particular case of our proposal.

B. PoROP-k using quadratic regression

However, as mentioned previously, *PoROP-k* methods allow to use more complex models. In this subsection we present the equations used to build a quadratic model, assuming that the resulting quadratic regression is $l_{m,n} = \gamma_m n^2 + \beta_m n + \alpha_m$ (where $n = 0 \ldots k-1$). Specifically, the expressions used to compute γ_m, β_m and α_m are presented in Figure 5. Analogously to the linear regression, these results can be derived from the normal equations.

IV. Experiments

In Section III, we have presented the *PoROP-k* protection methods, which protect a data set combining a new vision of the data to be protected with a complex pre-processing process and a model regression. In this section, we describe a set of experiments that allows us to test the new set of methods presented in this paper and compare them to *LiROP-k* methods.

A. Data

For evaluation purposes, we have considered the two reference data sets proposed in the CASC project [3]. The first has been extracted using the Data Extraction System (DES) from the U. S. Census Bureau [6], called Census. The second has been obtained from the U.S. Energy Information Authority [9], called EIA.

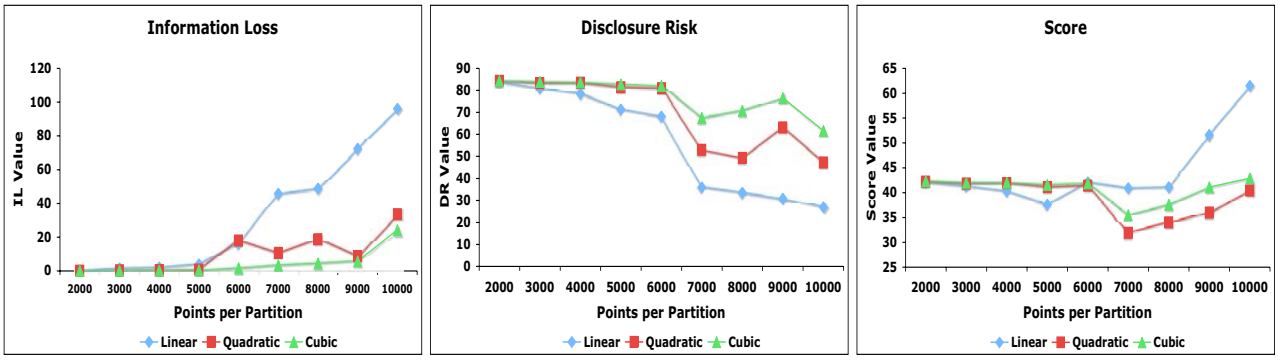
The Census data set contains 1080 records consisting of 13 attributes (which is equal to 14040 values to be protected). The *EIA* data set, after removing the identifiers and the categorical attributes, contains 4092 records consisting of 5 attributes. The total number of values to be protected in this data set is equal to 20460. For the experiments we assume that the intruder knows 7 out of the 13 attributes in the Census data set and 3 out of 5 attributes in the EIA data set.

B. Measures

We calculate the *score*, a typical general measure used to compare different protection methods [8], to evaluate *PoROP-k* methods. We also use this score to compare *PoROP-k* methods with *LiROP-k* methods.

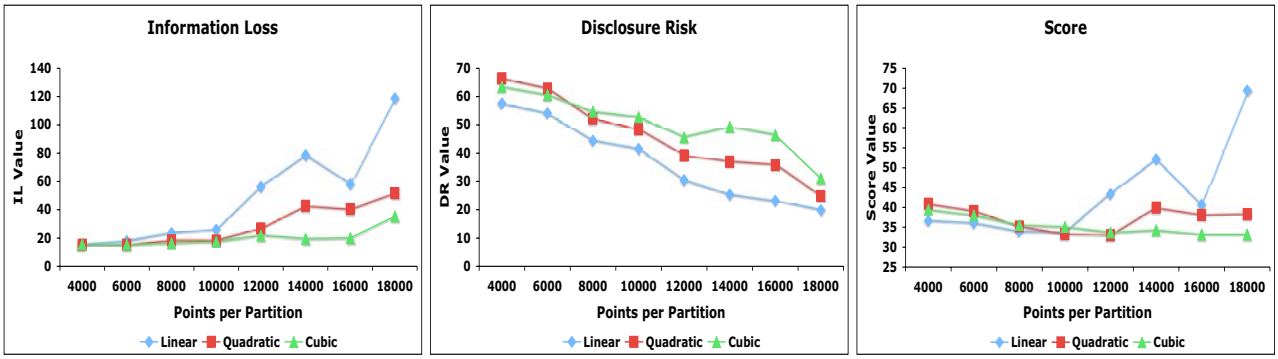
In order to calculate the *score*, we use the measures presented in previous work:

- **Information Loss (IL):** Let X and X' be matrices representing the original and the protected data set, respectively. Let V and R be the covariance matrix and the correlation matrix of X, respectively; let $\overline{X}$ be the vector of variable averages for X and let S be the diagonal of V. Define V', R',$\overline{X}'$, and S' analogously from X'. The information loss is computed by averaging the mean variations of $\overline{X}-\overline{X}'$,$V-V'$,$S-S'$, and the mean absolute error of $R-R'$ and multiplying the resulting average by 100. All these measures have been extracted from [8] and are computed in the same way.
- **Disclosure Risk (DR):** We use the three different methods presented in [11] in order to evaluate DR: (i) *Distance Linkage Disclosure risk* (DLD), which is the average percentage of linked records using distance based record linkage, (ii) *Probabilistic Linkage Disclosure risk* (PLD), which is the average percentage of linked records using probabilistic based record linkage and (iii) *Interval Disclosure risk* (ID) which is the average percentage of



(a) (b) (c)

Fig. 6. Average results of Information Loss (a), Disclosure Risk (b) and Score for the Census data set.



(a) (b) (c)

Fig. 7. Average results of Information Loss (a), Disclosure Risk (b) and Score for the Census data set.

original values falling into the intervals around their corresponding masked values. The three values are computed over the number of attributes that the intruder is assumed to know that, in our case, ranges from one to half of the attributes. These measures have been extracted from [8] and are computed in the same way:

$$DR = 0.25\,DLD + 0.25\,PLD + 0.5\,ID$$

- **Score:** A final score measure is computed by weighting the presented measures, also proposed in [8]:

$$score = 0.5\,IL + 0.5\,DR$$

C. Results

In order to understand whether using more complex regression methods allows us to preserve the information more accurately, we first study the information loss of each method.

We test *PoROP*$_s$-k and *PoROP*$_p$-k methods using linear, quadratic and cubic regressions. The range of values for the number of points per partition k has been defined in order to test a wide spectrum of cases ranging from low IL values to cases where the IL is high. Since the distribution of values in each data set is different, values of k are chosen specifically for each one, namely, for the Census data set k ranges from 2000 to 10000, while it ranges from 4000 to 18000 for the EIA data set.

We execute each configuration five times performing 510 tests in total. The average IL, DR and Score values for each configuration are presented in Figures 6 and 7. Tables I and II show the detail of the scores obtained from the experiments. Note that the figures and tables presented in this section only show the results using the synthetic version (*PoROP*$_s$-k). The results obtained by *PoROP*$_p$-k are almost identical and are omitted for the sake of simplicity.

In general, being able to control the IL is important, specially when we are interested in keeping the statistics in the protected data set. As we can see in Figure 6.(a) and Figure 7.(a), *PoROP*-k methods present a strong relation between IL and the value of parameter k. Usually, when parameter k increases, IL increases. Note that this can be observed independently of the model regression and the data set. In our case, the pre-processing phase is very important to guarantee this strong relation, since by vectorizing, ordering, partitioning and normalizing *PoROP*$_p$-k makes it possible to find a regression model that accurately fits the data set.

Observing the same figures, we can see that the more complex the polynomial model, the lower the information loss. This happens because by increasing the complexity of the regression function, we also increase the fitting capabilities

Census			
	Linear ($LiROP-k$)	*Quadratic*	*Cubic*
2000	42.1	42.2	42.3
3000	41.3	41.8	42.0
4000	40.3	41.9	42.0
5000	37.6	41.1	41.6
6000	42.1	41.4	41.9
7000	40.9	31.9	35.5
8000	41.1	34.0	37.6
9000	51.6	36.0	41.1
10000	61.5	40.4	42.9

TABLE I
AVERAGE SCORES FOR THE *PoROP$_s$-k* METHODS USING THE CENSUS DATA SET.

EIA			
	Linear ($LiROP-k$)	*Quadratic*	*Cubic*
4000	36.7	40.9	39.5
6000	36.1	39.1	38.0
8000	34.0	35.2	35.6
10000	33.8	33.3	35.1
12000	43.4	33.0	33.7
14000	52.1	39.9	34.3
16000	40.7	38.1	33.2
18000	69.4	38.3	33.2

TABLE II
AVERAGE SCORES FOR THE *PoROP$_s$-k* METHODS USING THE EIA DATA SET.

of the complex polynomial models.

A decrease in the information loss typically implies an increase in the disclosure risk. Figures 6.(b) and 7.(b) present the evolution of the disclosure risk as a function of k. We can observe that, as the complexity of the regression model increases, *PoROP-k* methods present a larger DR. However, as it follows, the overall decrease on the information loss compensate for the increase on the disclosure risk. This is shown in Figures 6.(c) and 7.(c), where we can observe that the score values obtained by quadratic and cubic regression models are in general lower than the values obtained by the linear regression models. This effect is more clear when partitions have a large number of points. In these cases *PoROP-k* methods outperform the results obtained by *LiROP-k* methods due to their greater fitting capabilities. Tables I and II show that, increasing the complexity of the regression functions, we achieve better quality in the protection. Specifically, in the Census data set, the best scores are obtained using quadratic regression (31.9), while the best scores using linear regressions are 37.6. Analogously, using the EIA data set the best scores are obtained using cubic regressions instead of linear regressions.

V. CONCLUSIONS AND FUTURE WORK

In this paper we have presented a generalization of the *LiROP-k* anonymizing methods, called *PoROP-k* methods. We have shown that the complexity of the regression model used to protect data is a relevant parameter and affects the overall quality of the protection method. In general, we have seen that, by making the model more complex, we can improve the quality of the protection method. Also, our new class of methods allows us to control the information loss by modifying the k parameter.

As future work, we plan to find new criteria to decide which is the best regression model for each partition in order to minimize the information loss preserving the disclosure risk as low as possible.

ACKNOWLEDGMENTS

The authors from UPC want to thank Generalitat de Catalunya for its support through grant number GRE-00352 and Ministerio de Educación y Ciencia of Spain for its support through grant TIN2006-15536-C02-02. Jordi Nin wants to thank the Spanish Council for Scientific Research (CSIC) for his I3P grant.

REFERENCES

[1] Adam, N. R., Wortmann, J. C., (1989), Security-Control for statistical databases: a comparative study. ACM Computing Surveys, Volume: 21, 515-556.
[2] Ackerman, M., Faith Cranor, L., Reagle, J., (1999), Privacy in e-commerce: examining user scenarios and privacy preferences, EC '99: Proceedings of the 1st ACM conference on Electronic commerce, ACM Press, ISBN: 1-58113-176-3, Pages: 1–8.
[3] Brand, R., Domingo-Ferrer, J., and Mateo-Sanz, J. M., (2002) Reference datasets 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, Volume: 13, 321-327.
[5] Dahlquist, G., Björck, A., (2003), Numerical methods, Mineola, Dover Publications.
[6] Data Extraction System, U.S. Census Bureau, http://www.census.gov/
[7] Domingo-Ferrer, J., Torra, V., (2001), Disclosure Control Methods and Information Loss for Microdata, Confidentiality, Disclosure, and Data Access: Theory and Practical Applications for Statistical Agencies, Elsevier Science, 91-110.
[8] 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, Elsevier Science, 111-133.
[9] U.S. Energy Information Authority, http://www.eia.doe.gov/
[10] Medrano-Gracia, P., Pont-Tuset, J., Nin, J., Muntés-Mulero, V., (2007), Ordered Data Set Vectorization for Linear Regression on Data Privacy, Lecture Notes in Computer Science, Springer. To be published.
[11] Torra, V., Domingo-Ferrer, J., (2003), Record linkage methods for multidatabase data mining, Information Fusion in Data Mining, Springer, 101-132.