WP. 13
ENGLISH ONLY

UNITED NATIONS STATISTICAL COMMISSION and
ECONOMIC COMMISSION FOR EUROPE
CONFERENCE OF EUROPEAN STATISTICIANS

EUROPEAN COMMISSION
STATISTICAL OFFICE OF THE
EUROPEAN COMMUNITIES (EUROSTAT)

Joint UNECE/Eurostat work session on statistical data confidentiality
(Bilbao, Spain, 2-4 December 2009)

Topic (ii): Synthetic and hybrid data

DEALING WITH EDIT CONSTRAINTS IN MICRODATA PROTECTION: MICROAGGREGATION

Invited Paper

Prepared by Vicenç Torra, Isaac Cano and Guillermo Navarro-Arribas, IIIA-CSIC, Catalonia, Spain

Dealing with edit constraints in microdata protection: microaggregation

Vicenç Torra, Isaac Cano, Guillermo Navarro-Arribas

IIIA - Articial Intelligence Research Institute,
CSIC - Spanish Council for Scientic Research,
Campus UAB s/n, 08193 Bellaterra (Catalonia, Spain)
({vtorra, cano, guille}@iiia.csic.es)

Abstract.

In this paper we discuss how most edit constraints can be taken into account in an effective way through microaggregation. We discuss different edit constraints and some variations of microaggregation that permits to deal with such constraints. We will also present our software to formalize and deal with such constraints in an automatic way.

1 Introduction

When perturbation methods are used to protect statistical data they can introduce undesirable errors in the data. For instance, data editing [9, 12, 4] is a field of statistical disclosure control that is devoted to the analysis and correction of raw data for their improvement. The basic idea is that data should satisfy a set of requirements (or constraints) before their release. E.g. non negative values are not permitted for people's age. Data editing is typically applied to the original data and, in any case, before any perturbation takes place. Thus, the perturbation can introduce inconsistency in the data.

The study of perturbation methods in the presence of data edits has not been considered until very recently [16, 14]. We provide a discussions about some data edits and how can they be preserved in microaggregation. We also describe a framework to automate the microaggregation of constrained data, which uses XML as the base format both for the microdata to be processed and to express the edit constraints. The framework identifies the required edit constraints and applies modifications of the microaggregation method in order to perturb the data while satisfying the edit constraints.

The structure of the paper is as follows. Section 2 provides an overview of microaggregation, and Section 3 discusses the data edits in microaggregation. Section 4, and 5 presents our implementation and results, and Section 6 concludes the paper.

2 Overview of microaggregation

In this paper we show how microaggregation [3] can cope with edit constraints.

From the operational point of view, microaggregation is defined in terms of partition and aggregation:

- **Partition.** Records are partitioned into several clusters, each of them consisting of at least k records.
- **Aggregation.** For each of the clusters a representative (the centroid) is computed, and then original records are replaced by the representative of the cluster to which they belong to.

In most cases microaggregation is applied to numeric data, even so, it can also be applied to categorical data [15], either nominal or ordinal [8]. Moreover microaggregation normally considers a crisp partition of the records (as the k-means clustering), but there is also some works that do consider the use of fuzzy c-means to partition the dataset [19], and then aggregate the records accordingly. Although our work is mainly focused on crisp clustering with numeric data, we will also consider other possibilities if appropriate.

In the rest of the paper we will use the following notation. We consider a microdata file with n records $x_1, \ldots, x_n$ that take values over a set of variables $V_1, \ldots, V_m$. We express the value for record x_i in variable V_j by $x_{i,j}$.

The function $\mathbb{C}$ is the cluster representative or centroid, which we assume to be a function of the data in the cluster. More specifically, we presume that the representative of the variable V is a function of the values of the records for V, that is, $\mathbb{C}(x_1, \ldots, x_N)$. Similarly, the representative for variable V_i is $\mathbb{C}(x_{1,i}, \ldots, x_{N,i})$.

Note that in most cases edit constraints are preserved by providing a specific function $\mathbb{C}$, which preserves the constraint.

3 Data editing and microaggregation

Data editing can broadly be defined as the process of detecting errors in statistical data [2]. In general the whole data editing process can be very costly, even requiring human supervision in some stages [4]. For this reason it is very desirable that the statistical disclosure control methods used in edited data do not introduce new errors, so data does not need to be edited again.

The editing process is usually formalized as a set of edit constraints, that the data should satisfy. We present the generic classification of edit constraints from [16], and show their applicability in a slightly modified version of the Census data set [1]. The modification of the dataset is minimal and restricted to the addition of three variables in order to be able to show the applicability of the types of data edits.

We depart from an XML representation of the microdata. Our microdata file has a simple generic format, where data are stored by rows, and each value is an element labeled with the variable name. It resembles most of current XML standards for data spreadsheets such as Office OpenXML or OpenDocument. The simplicity of this format and the availability of a great number of tools for XML processing makes it easy to obtain it from not only other similar XML files but also directly from database tables, or more generic microdata files.

In order to provide a standardized and already in-use language to represent the edit constraints we have used Schematron [10], which is a rule-based validation language for XML. Unlike common schema languages for XML such as W3C Schema, RELAX NG, or DTDs, which can express rules about the structure of the document, Schematron also provides semantic rules, which makes it very suitable to express edit constraints.

Schematron expresses pattern rules both as *assert*s or *report*s. An *assert* tags positive assertions about the document, while the *report* tags negative assertions. The assertions themselves are declared as the attribute *test* with an XPath [20] expression. We have chosen to express edit constraints as Schematron *assert*s, because they do more clearly express the semantic of the constraint, but *report*s could also be used to achieve the same effect. These Schematron rules can be easily checked against the XML microdata file directly and automatically by an XSLT engine.

What follows is a description of the different data edit constraints, how are they encoded as Schematron rules[1], and how microaggregation can be applied to preserve the constraints.

3.1 Linear constraints (EC-LC)

A variable can be expressed as a linear combination of a set of other variables. For example, the following relation between *family income*, *person income*, and *other persons income* should hold (cf. Fig. 1):

$$\begin{aligned}\text{EC-LC: } & \textit{other person income} + \textit{person income} \\ & = \textit{family income} \\ & \Rightarrow V9 + V10 = V14\end{aligned}$$

A linear constraint can be expressed, if we assume that V is the dependent variable, as $V = \sum_{i=1}^{K} \alpha_i V_i$, for some values α_i and variables V_i.

Assuming that the original data (already edited) satisfies the linear constraint, so $x_j = \sum_{i=1}^{K} \alpha_i x_{j,i}$, we need to consider which function is suitable for computing the cluster representative. The most general solution for $\mathbb{C}$ in this case is,

$$\mathbb{C}(x_1, \ldots, x_N) = \frac{1}{N} \sum_{i=1}^{N} x_i \tag{1}$$

[1] Note that in these examples, the XPath expression in the test attribute, uses the entities '&lt;', and '&gt;' as the symbols '<', and '>', because the expression is contained in a string.

```
<pattern name="EC-LC: V9+V10 = V14">
  <rule context="Record">
    <assert test="1.0 * number(V9)
                 + 1.0 * number(V10)
                 = number(V14)">
    Linear Constraint
      total_others <value-of select="V9"/>
     + total_personal <value-of select="V10"/>
     != total_family <value-of select="V14"/>
    </assert>
  </rule>
</pattern>
```

Figure 1: Schematron rule for EC-LC.

```
<pattern name="EC-NLC: V5*V15 = V16">
  <rule context="Record">
    <assert test="number(V5)
                 * number(V15)
                 = number(V16)">
    Non-linear Constraint
     fed. income tax <value-of select="V5" />
      * inv. state income tax
      <value-of select="V15" />
      != fed./state ratio
      <value-of select="V16" />
    </assert>
  </rule>
</pattern>
```

Figure 2: Schematron rule for EC-NLC.

Note that it coincides with the arithmetic mean.

Preservation of linear constrains in fuzzy microaggregation (microaggregation based on fuzzy clustering algorithms), can also be achieved. In [17], the authors provide a fuzzy c-means algorithm, which preserves linear constraints.

3.2 Non-linear constraints (EC-NLC)

Some numerical variables satisfy a non-linear relation. For example (cf. Fig. 2):

$$\begin{aligned}\text{EC-NLC: } & \textit{fed. inc. tax} * \textit{inv. state inc. tax} \\ & = \textit{ratio fed.-state inc. tax} \\ & \Rightarrow V5 * V15 = V16\end{aligned}$$

In this case we can follow the same approach considering multiplicative variables. Formally, we consider variables V, V_1, ..., V_K satisfying $V = \prod_{i=1}^{K} V_i^{\alpha_i}$. In this case the most general solution for $\mathbb{C}$ is,

$$\mathbb{C}(x_1, \ldots, x_N) = \prod_{i=1}^{N} x_i^{1/N} \tag{2}$$

Note that it coincides with the geometric mean.

3.3 Constraints on the possible values (EC-PV)

The values of a given variable are restricted to a predefined set. For example, stating that the value of variable *employer contribution for health care* should be in the interval [0, 7500] (cf. Fig. 3).

$$\text{EC-PV:}\textit{employer contrib. health} \in [0, 7500] \Rightarrow V3 \in [0, 7500]$$

Or for example, consider an attribute *age* where a value of 18.5 may not make sense, and only integer positive values are permitted. Other similar constraint could involve subsets of variables, which could be reformulated in similar terms.

```
<pattern name="EC−PV: V3 in [0, 7500]">
  <rule context="Record">
    <assert test="0 &lt;= number(V3) and
                    number(V3) &lt;= 7500">
      Constraints on possible values
      Employer contribution for health care
      <value−of select="V3"/>
      is not in the interval [0, 7500]
    </assert>
  </rule>
</pattern>
```

Figure 3: Schematron rule for EC-PV

```
<pattern name="EC−LC: if V8 &lt; 1115 then
               V13 &lt;= V12">
  <rule context="Record">
    <assert test="not(number(V8) &lt; 1115) or

(number(V13) &lt;= number(V12))">
      IF total person earnings
      <value−of select="V8"/>
      &lt; 1115 THEN
      total wage and salary
      <value−of select="V13"/>
      &lt;= taxable income
      <value−of select="V12"/>
      does not hold
    </assert>
  </rule>
</pattern>
```

Figure 4: Schematron rule for EC-GV.

In order to enforce constrains on the possible values, we can require the cluster representative to be in the interval defined between the minimum and the maximum of the elements in the cluster, that is, it has to satisfy *internality.* Formally,

$$\min_i x_i \leq \mathbb{C}(x_1, \ldots, x_N) \leq \max_i x_i$$

Note that if the constraint is that $x_i \in [a, b]$ for some a and b, it is clear that for edited data, we have $x_i \in [a, b]$, and thus, this constraint implies that $\mathbb{C}(x_1, \ldots, x_N) \in [a, b]$. It can be proved that both Eq. (1) and Eq. (2) do satisfy internality [16].

This constrain is commonly found in categorical nominal data. E.g. $vehicle \in \{car, motorcycle, truk, \ldots\}$. By using the plurality rule (or mode) as the function $\mathbb{C}$, this constraint is preserved. This aggregator (which can be generalised as the weighted plurality rule) selects the most frequent element from the cluster.

The microaggregation of categorical ordinal data preserving this constraint can also be achieved, by using the median (also the weighted median, or convex median) as the function $\mathbb{C}$.

$$\mathbb{C}(x_1, \ldots, x_N) = \begin{cases} x_{\sigma(\lfloor (N+1)/2 \rfloor)} & \text{if } N \text{ is even} \\ x_{\sigma((N+1)/2)} & \text{if } N \text{ is odd} \end{cases}$$

where $\{\sigma(1), \ldots, \sigma(N)\}$ is a permutation of $\{1, \ldots, N\}$ such that $x_{\sigma(i-1)} \leq x_{\sigma(i)}$ for all $i = \{2, \ldots, N\}$.

Both the plurality and median operators, which satisfy internality, are commonly used in the microaggregation of categorical data [7].

3.4 One variable governs the possible values of another one (EC-GV)

The values of a variable are constrained by the values of another one. E.g., considering the relations between three variables *total person earnings*, *taxable income* and *amount* as (cf. Fig. 4):

$$\begin{aligned} \text{EC-GV: IF } & \textit{total person earnings} < 1115 \\ & \text{THEN } \textit{taxable income} \leq \textit{amount} \\ & \Rightarrow \text{ IF } V8 < 1115 \text{ THEN } V13 \leq V12 \\ & \Rightarrow \textit{not}(V8 < 1115) \text{ or } (V13 \leq V12) \end{aligned}$$

In general any monotonic function $\mathbb{C}$, permits us to generate a protected file with $V_1 < V_2$ for variables V_1 and V_2 if in the original file it also holds $V_1 < V_2$. In fact, $x_{i,j} \leq x_{i,k}$ for all i and $j \neq k$ implies $\mathbb{C}(x_{1,j}, \ldots, x_{N,j}) \leq \mathbb{C}(x_{1,k}, \ldots, x_{N,k})$, corresponds to the monotonicity of $\mathbb{C}$. Note also that Eq. (1) and Eq. (2) are monotonic. Simple EC-GV constraints, such as: EC-GV1 : $V_3 \leq V_7$ are satisfied by using a monotonic function $\mathbb{C}$.

In the case of categorical data, this constraint only makes sense in ordinal data, and the median is a monotonic function.

Other EC-GV constraints such as the one presented in Section 3, which can be summarized as:

$$\text{IF } V8 < 1115 \text{ THEN } V13 \leq V12$$

can be satisfied by partitioning the dataset in subsets according to the antecedent in the rule, and then applying microaggregation separately to each subset using a monotonic function $\mathbb{C}$. In this case the data is partitioned in two sets, one with records satisfying $V8 < 1115$, and the other with records with $V8 \geq 1115$. Note that the same strategy works for categorical ordinal data.

3.5 Other types

Other classes of constraints might be considered. For example, constraints on non-numerical variables (ordinal or categorical), ...

If we consider data editing in the context of perturbative statistical disclosure control an additional constraint is normally assumed.

- *Values are restricted to exist in the domain.* Not only the values should belong to a predefined set (as in EC-PV constraints), but the values should really exist in the domain. For example due to the edit constraint EC-PV previously presented, the variable *employer contrib. health* has to be in the interval $[0, 7500]$, but if in the original data all values are under 500 the perturbation introduced in the masked data cannot cause a record to have a value of 800. Another example can be an attribute with the *town o village of residence* of the individual. The protected microdata cannot introduce for example a town that was not in the original microdata. (cf. Fig. 5)

```
<pattern name="Value−domain restriction">
  <rule context="Record">
    <let name="original"
         value="document('microdata.xml')" />
    <assert test="exists(index−of
                  ($ original //Record/V1, V1))">
      Value <value−of select="V1" />
      is not in domain of original values for V1
    </assert>
  </rule>
</pattern>
```

Figure 5: Schematron rule for 'values are restricted to exist in the domain'.

An appropiate operator for $\mathbb{C}$ that satisfies this constraint is the *median*, which has already been used for microaggregation in [13]. Other operators such as order statistics and boolean max-min functions [19] could also be used.

The median (as well as the order statistics) are monotonic functions. Due to this, they could also be applied in the case of constraints where one variable govers the possible values of another one (EC-GV). This monotonicity makes the median also suitable for constraints on the possible values (EC-PV). Regarding the other constraints, linear and non-linear, it is important to note that the functions introduced in Eq. (1) and Eq. (2) cannot deal with this constraint.

Note that in this case the operators discussed for categorical data (mode, and median) satisfy this constraint.

4 Implementation details

The constrained microaggregation has been implemented with the aim of providing an automated process to microaggregate edited data.

The original microdata from the Census datase is processed together with the specification of the data edits as Schematron rules, and then the data is microaggregated according to the edit constraints. Finally, edit constraints can be checked in the masked file to verify the edit constraints. Note that we have only considered numerical data.

As usual in microaggregation, variables are microaggregated by groups. In our case, all variables involved in an edit constraint are grouped together. Note that for constraints EC-PV, and EC-GV, both the arithmetic mean and the geometric mean can be used as the cluster representative function. We use the arithmetic mean in these cases because it yields better results regarding the information loss of the protected data.

5 Results

We have considered two scenarios. In scenario $S1$ we have microaggregated the file considering all edit constraints and the remaining variables, the ones that are not involved in any edit constraint, are microaggregated together grouping them in groups of size 3. In the scenario $S2$ we have microaggregated the whole dataset without taking into account the edit constraints using the arithmetic mean to compute $\mathbb{C}$ and again making groups of 3 variables.

For each scenario we have measured its utility and protection. To compute the information loss the Probabilistic Information Loss [11] (PIL) measure has been used. On the other hand, to compute the disclosure risk(DR), it was taken into account two broadly used measures, the Distance Based Record Linkage (DBRL) and the Interval Disclosure (ID) [5, 6]. Hence the average disclosure risk is the arithmetic mean of $DBRL$ and ID. Finally, the $SCORE$ is computed as a mean of the PIL and DR.

The experiments have shown that microaggregating while considering the edit constraints slightly affects the information loss and disclosure risk. Usually, the desired value of k is taken between 4 and 10. As it can be seen, in our case, for such values of k the *SCORE* values are very similar. Although in *S1* the minimum *SCORE* is 37.223 for $k = 10$, all other *SCORE* values for the range $k \in [4, 10]$ are closely similar. The same applies to scenario *S2*, where the minimum *SCORE* is 34.531 for $k = 6$. Note that the lower score, the better, and that only scores under 50 are of interest (this is the score of unprotected data). The difference between both minimum values of the *score* is compensated with a preservation of the edit constrains in the original and masked dataset in case of scenario $S1$. Moreover, a score of 37.223 (from scenario *S1*) is considered a good one, providing a proper trade off between information loss and disclosure risk.

To get a graphical representation of *PIL*, *DR*, and *SCORE* we have plotted in the Fig.s 6 and Fig. 7 their relationship for all k values from 1 to 99. In these figures it is shown that the *score* remains almost constant because of the greater the information loss the lower the disclosure risk in almost the same proportion.

6 Conclusions

In this paper we have proposed a new framework for the automatic perturbation of data through microaggregation taking into account the requirements or the constraints that the data elements have to satisfy. We have assessed the information loss and disclosure risk when considering or not the edit constraints in the microag-

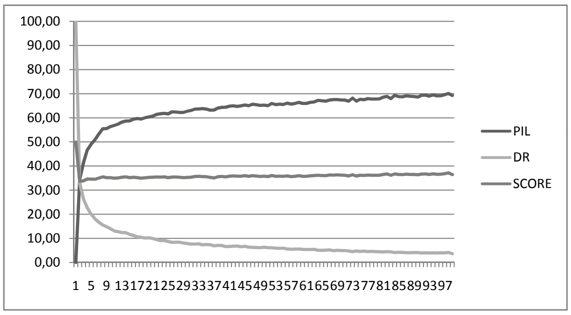

Figure 6: Scatter plot showing PIL and DR with respect to group's size k for scenario $S1$.

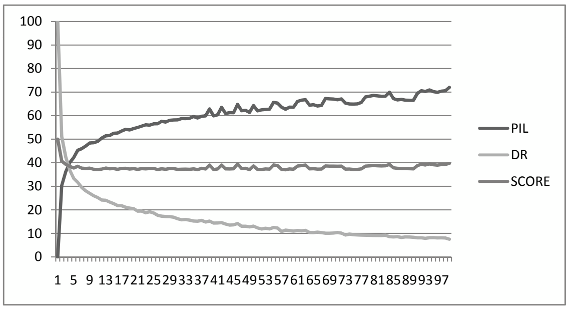

Figure 7: Scatter plot showing the relationship between PIL and DR with respect to group's size k for scenario $S2$.

gregation process. We have presented the results when microaggregating taking into account 4 different types of edit constraints, $EC-PV$, $EC-GV$, $EC-LC$ and $EC-NLC$. As future work we consider the extension of the approach to deal with categorical attributes and to support more edit constraints.

Acknowledgments

Partial support by the Spanish MICINN (projects eAEGIS TSI2007-65406-C03-02, ARES - CONSOLIDER INGENIO 2010 CSD2007-00004, and TSI2006-03481), and the "Institut d'Estadística de Catalunya (IDESCAT)" is acknowledged.

References

[1] U.S. Census Bureau. Data Extraction System. http://www.census.gov/.

[2] Chambers, R. Evaluation criteria for statistical editing and imputation. *National Statistics Methodological* series No.28, Jan 2001.

[3] Deejay's, D., Nanopoulos, P. Panels of enterprises and confidentiality: the small aggregates method, in *Proc. of 92 Symposium on Design and Analysis of Longitudinal Surveys, Statistics Canada*, 1993, pp. 195–204.

[4] De Waal, T. An overview of statistical data editing. Statistics Netherlands. 2008.

[5] 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, pp. 111 – 133.

[6] 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, pp. 91 – 110.

[7] Domingo-Ferrer, J., Torra, V., (2005), "Ordinal, continuous and heterogeneous k-anonymity through microaggregation," *Data Mining and Knowledge Discovery*, vol. 11, no. 2, pp. 195 – 212.

[8] Domingo-Ferrer, J., Torra, V. Ordinal, continuous and heterogeneous k-anonymity through microaggregation, *Data Mining and Knowledge Discovery*, pp. 195–212, Jan. 2005.

[9] Granquist, L. The new view on editing, *Int. Statistical Review* 65:3 381-387. 1997.

[10] ISO/IEC. Information technology – Document Schema Definition Language (DSDL) – Part 3: Rule-based validation – Schematron. ISO/IEC 19757-3:2006 Standard JTC1/SC34, 2006.

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

[12] Pierzchala, M. A review of the state of the art in automated data editing and imputation, in *Statistical Data Editing*, Vol. 1, Conference of European Statisticians Statistical Standards and Studies N. 44, UN Statistical Commission and Economic Commission for Europe, 10-40. 1995

[13] Sande, G. Exact and approximate methods for data directed microaggregation in one or more dimensions, *Int. J. of Unc., Fuzz. and Knowledge Based Systems* 10:5 459–476. 2002.

[14] Shlomo, N., De Waal, T. Protection of Micro-data Subjec to to Edit Constraints Against Statistical Disclousure. *Journal of Official statistics.* Vol. 24, No. 2, pp. 229–253. 2008.

[15] Torra, V. Microaggregation for categorical variables: A median based approach, in *Proc. Privacy in Statistical Databases (PSD 2004)*, ser. LNCS, vol. 3050, Jan. 2004, pp. 162–174.

[16] Torra, V. Constrained microaggregation: Adding constraints for data editing, *Transactions on Data Privacy*, vol. 1, no. 2, pp. 175–186, 2008.

[17] V. Torra, On the Definition of Linear Constrained Fuzzy c-Means, *Proc. of the EUROFUSE 2009* Sep. 2009, pp. 61-66.

[18] Torra, V., Miyamoto, S. Evaluating fuzzy clustering algorithms for microdata protection. in *Proc. Privacy in Statistical Databases (PSD 2004)*, ser. LNCS, vol. 3050, Jan. 2004, pp. 162–174.

[19] Torra, V., Narukawa, Y. *Modeling decisions: information fusion and aggregation operators*, Springer. 2007.

[20] W3C. XML Path Language (XPath) 2.0. W3C Recommendation, Jan. 2007.