UNITED NATIONS STATISTICAL COMMISSION and WP.16
ECONOMIC COMMISSION FOR EUROPE ENGLISH ONLY

CONFERENCE OF EUROPEAN STATISTICIANS

Work Session on Statistical Data Editing
(Bonn, Germany, 25-27 September 2006)

Topic (iii): Editing Microdata for Release

STATISTICAL DISCLOSURE CONTROL FOR CONTINUOUS VARIABLES UNDER EDIT RESTRICTIONS

Invited Paper

Submitted by Southampton Statistical Sciences Research Institute, University of Southampton and Statistics Netherlands[1]

Abstract: Before releasing statistical outputs, data suppliers have to assess the risk of disclosing confidential information about the statistical units and apply Statistical Disclosure Control (SDC) methods if necessary. SDC methods perturb, modify or summarize the data depending on the format for releasing the statistical data, i.e. microdata or tabular data. The goal is to choose an optimal SDC method that manages disclosure risk below a tolerable risk threshold while ensuring high utility and high quality statistical data. In particular, no edit failures should reoccur in the statistical data as a result of the perturbation process since this not only affects the utility of the data but also targets records that were perturbed making them susceptible to potential attacks.

In an earlier paper, we focused on a perturbation method for categorical variables based on a post-randomization probability mechanism (PRAM) and discussed its impact on the logical consistency of records in microdata prior to its release and implementation methods for minimizing loss of utility. In this paper we examine and generalize SDC methods for continuous data. The methods all perturb the data in some way. We develop methods of perturbation, which minimize both micro-edits (i.e. record level edit failures) and macro-edits (i.e. overall measures which assess information loss and utility) and also manage the disclosure risk.

I. INTRODUCTION

1. Many Statistical Agencies have provisions for releasing microdata from social surveys for research purposes usually under special license agreements and through secure data archives. Microdata from business surveys are typically not released because of their disclosive nature due to high sampling fractions and skewed distributions. In order to preserve the privacy and confidentiality of individuals responding to social surveys, Statistical Agencies must assess the disclosure risk in microdata and if required choose appropriate Statistical Disclosure Control (SDC) methods to apply to the data (see also Willenborg and De Waal, 2001, Elamir and Skinner, forthcoming, Skinner and Shlomo, 2005, Rinott and

[1]Prepared by Natalie Shlomo (n.shlomo@soton.ac.uk) and Ton De Waal (twal@cbs.nl)

Shlomo, forthcoming, and Shlomo and De Waal, 2005). Measuring disclosure risk for the SDC decision problem involves assessing and evaluating numerically the risk of re-identifying statistical units. Disclosure risk typically arises from attribute disclosure where small counts on cross-classified indirect identifying key variables (such as: age, sex, place of residence, marital status, occupation, etc.) can be used to identify a statistical unit and confidential information may be learnt. SDC methods perturb, modify, or summarize the data in order to prevent re-identification by a potential attacker. Higher levels of protection through SDC methods however impact negatively on the utility and quality of the data. The SDC decision problem involves finding the optimum balance between managing disclosure risk to tolerable thresholds depending on the mode for accessing the data and ensuring high utility in the data.

2. Microdata that has had all of its records checked and corrected for logical inconsistencies through edit and imputation at the data processing phase may start failing edits as a result of the SDC methods applied. Therefore, we define the term **post-editing** to refer to the edit and imputation process that is carried out before disseminating statistical disclosure controlled microdata. In Shlomo and De Waal, 2005, two types of edits are defined in post-editing: micro edits which refer to logical inconsistencies at the record level similar to the edit constraints defined for the edit and imputation carried out at the data processing stage; and macro edits which refer to overall loss of utility after applying SDC methods to microdata. Macro edits involve minimizing quantitative measures for assessing the effects on statistical inference: the impact on bias and variance of estimates, distortions to distributions, effects on goodness of fit criteria for statistical modeling, etc. Both types of edits need to be assessed before releasing the microdata. Micro edits need to be corrected, not only to ensure that no inconsistencies remain in the microdata at the record level, but also to avoid potential intruders from pinpointing individuals and their variables that were perturbed in order to decode the SDC methods applied. Macro edits cannot be corrected entirely but should be minimized in order to ensure high utility and fit for purpose data.

3. SDC methods for microdata include perturbative and non-perturbative methods. Non-perturbative methods preserve the integrity of the data by limiting the amount of information released in the microdata without actually altering the data. These methods include sub-sampling, local suppression, recoding and coarsening variables, and eliminating variables. With these methods, the logical consistency of the records remains unchanged and micro edits will not begin to fail as a result of these SDC methods. It is important, nevertheless, to assess macro edits and whether the resulting disclosure controlled microdata set is fit for purpose. Perturbative methods however alter the data, and therefore we expect consistent records to start failing micro edits due to the perturbation. Perturbative methods for continuous variables include rounding to a preselected rounding base, microaggregation (replacing values with their average within groups of records), adding random noise, and rank swapping (swapping values between pairs of records within small groups). Perturbative methods for categorical variables include record swapping (typically swapping geography variables) and a more general post-randomization probability mechanism (PRAM) where categories of variables are changed or not changed according to a prescribed probability matrix and a stochastic selection process. For perturbative methods there is a need to carry out post-editing on the released and disclosure controlled microdata, both on the micro level and on the macro level.

4. In this paper we focus on some common perturbative methods for SDC for continuous variables and how they alter the data and affect the consistency and quality of the microdata. We demonstrate some of these methods on a microdata set from the 2000 Israel Income Survey. We suggest ways of minimizing the number of failed micro edits as a result of the perturbation as well as improve some of the macro edits through innovative techniques for applying the SDC methods. Sections II through V describe the SDC methods under analysis that will be addressed in this paper: additive noise, rounding, microaggregation, and rank swapping. We close in Section VI with a discussion.

II. ADDITIVE NOISE

5. Additive noise is an SDC method that is carried out on continuous variables. Random noise is generated identically independently distributed with a mean of zero in order to ensure that no bias is introduced into the original variable and a positive variance. The random noise is then added to the original variable. It has been shown that re-identification can occur using this SDC method based on probabilistic record linkage techniques (Yancey, Winkler and Creecy, 2002). This has led towards some development of mixture models for generating random noise, which achieve higher protection levels. Adding random noise will not change the mean of the variable but may introduce more variance for the estimate of the mean of the variable. This will impact on the ability to make statistical inference, particularly for estimating parameters in a regression analysis. In this section we examine several methods for adding random noise, which focus on preserving micro and macro edits.

6. Adding noise to a variable such as income may cause micro edit failures at the record level. For example, consider the micro edits:

E1a: gross income (*gross*) $\geq$ 0,

E1b: net income (*net*) $\geq$ 0,

E1c: taxes (*tax*) $\geq$ 0

and

E2: IF age $\leq$ 17 THEN gross income $\leq$ mean income.

Adding noise across the whole file may cause these edits to start failing. For example, in the 2000 Israel Income Survey, out of 32,896 individuals aged 15 and over surveyed, 16,232 individuals earned an income from salaries. The mean of their gross income from salaries was 6,910 IS with a standard deviation of 7,180 IS. Random noise is generated using a normal distribution with a mean of 0 and a variance that is 20% of the variance of the income variable ($0.2 \times (7{,}180^2)$). After adding the random noise to the income variable *gross*, 1,685 individuals failed the non-negativity edit E1a and out of 119 individuals under the age of 17, 6 individuals failed edit E2. It is clear that more control should be placed into the perturbation scheme in order to minimize the number of failed micro edits. We can generate, for example, random noise for each strata defined by quintiles of gross income as follows: sort the file according to the variable *gross*; define 5 equal groupings (i.e., quintiles); generate random noise separately in each quintile using 20% of the variance of the variable *gross* in the quintile as described above. Based on this method, we obtain that now only 66 individuals fail the non-negativity edit E1a and no individuals under the age of 17 fail edit E2. Moreover, based on the first method using the overall variance of the variable *gross*, the resulting perturbed variable had a standard deviation of 7,849 compared to 7,180. However, when perturbing the variable *gross* within quintiles, this led to an increase in the standard deviation of only 7,487.

7. In order to gain utility at a macro level, we can also carry out a method for generating additive random noise that is correlated with the variable to be perturbed, thereby ensuring that not only are means preserved but also the variance. Some methods for generating correlated random noise have been discussed in the literature based on transformations and fixed parameters (Kim, 1986, Fuller 1993, Brand, 2002, Yancey, Winkler and Creecy, 2002). We propose, however, an alternative method for generating correlated random noise that preserves means and variances that is very easy to implement. We demonstrate our method first on the univariate income variable *gross* as follows:

- Define δ which controls the amount of random noise added and calculate: $d_1 = \sqrt{(1-\delta^2)}$ and $d_2 = \sqrt{\delta^2}$.
- Generate random noise ε with a mean of $\mu' = \frac{1-d_1}{d_2}\mu$ where $\mu = 6{,}910$ is the mean of the original income variable, and a variance $\sigma^2 = 7{,}180^2$ of the original income variable.
- Calculate the perturbed variable: $gross' = d_1 \times gross + d_2 \times \varepsilon$
- Note that $E(gross') = d_1 E(gross) + d_2[\frac{1-d_1}{d_2} E(gross)] = E(gross)$ and $Var(gross') = (1-\delta^2)Var(gross) + \delta^2 Var(gross) = Var(gross)$.

As defined in paragraph 6, the above method can also be carried out within quintiles in order to minimize the number of micro edit failures. Indeed, based on this method within quintiles and using $\delta = 0.3$ (which is similar to the amount of noise generated in paragraph 6), we obtain that now only 9 individuals fail the non-negativity edit E1a and no individuals under the age of 17 fail edit E2. Moreover, the overall standard deviation of the perturbed variable has remained unchanged with a value of 7,198.

8. An additional problem when adding random noise is that there may be several variables to perturb at once, and these variables may be connected through an edit constraint. Consider for example in the 2000 Israel Income Survey three variables: gross income from salaries, taxes and net income from salaries. The original microdata set that has undergone edit and imputation processing will have ensured that no records fail the following edit:

E3: net income + taxes = gross income.

However, after perturbing each variable separately, this edit constraint will not be guaranteed. Therefore, we can split this procedure into two separate processes: (1) first carry out the perturbation method of adding random noise on each of the variables; (2) implement an additional stage of post-editing for correcting the additivity of the variables based on linear programming under the minimum change paradigm. Let the number of continuous variables be given by *n*. Denote the perturbed continuous variables after the first step by x_i (*i*=1,…,*n*) and the adjusted perturbed continuous variables by $\hat{x}_i$ (*i*=1,…,*n*). The linear programming problem for the second step can then be formulated as

minimize $\sum_i w_i \mid x_i - \hat{x}_i \mid$,

subject to the constraint that the $\hat{x}_i$ (*i*=1,…,*n*) satisfy all edits. Here the w_i (*i*=1,…,*n*) are non-negative weights expressing how serious a change in the *i*-th perturbed value is considered to be. In our case, we perturb variables *tax* and *net*, so *n* = 2. The constraints are based on: non-negativity (edits E1a, E1b and E1c), and additivity to the fixed total (*gross*). We also aim to preserve the ratio x_1/x_2 before and after the adjustments, where x_1 denotes the value of *tax* and x_2 the value of *net*. Aiming to preserve the ratio x_1/x_2 before and after perturbation gives us another linear constraint. The resulting linear programming problem can easily be solved by means of the EXCEL solver. However, in our case we can avoid the need for linear programming altogether for this specific problem, because owing to the structure in the data and the correlations between the variables, the algorithm based on paragraph 7 for generating multivariate normally distributed correlated random noise will result in the noise variables themselves preserving the additivity (see also paragraph 9 below). Therefore, when combining the noise in a linear combination with the original values of the variables, the additivity will be maintained.

9. The following describes how to generate the multivariate normally distributed noise vector which preserves means, covariances and additivity:

- Generate multivariate random noise in each quintile (note that we drop the index for quintile), using a standard SAS macro: $\begin{pmatrix} \varepsilon_{(GROSS)} \\ \varepsilon_{(NET)} \\ \varepsilon_{(TAX)} \end{pmatrix} \sim N(\boldsymbol{\mu}', \boldsymbol{\Sigma})$. The vector $\boldsymbol{\mu}'$ contains the corrected means of each of the three variables: gross income, net income and taxes:
  $\boldsymbol{\mu}'^{\mathrm{T}} = (\boldsymbol{\mu}'_{(GROSS)}, \boldsymbol{\mu}'_{(NET)}, \boldsymbol{\mu}'_{(TAX)}) = (\frac{1-d_1}{d_2}\boldsymbol{\mu}_{(GROSS)}, \frac{1-d_1}{d_2}\boldsymbol{\mu}_{(NET)}, \frac{1-d_1}{d_2}\boldsymbol{\mu}_{(TAX)})$. The matrix $\boldsymbol{\Sigma}$ is the covariance matrix where the diagonals contain the original variances of the three variables in each quintile: gross income, net income and taxes, and the off-diagonals contain their covariances:
  $$\boldsymbol{\Sigma} = \begin{pmatrix} \sigma^2_{GROSS} & \rho_{(GROSS,NET)}\sigma_{(GROSS)}\sigma_{(NET)} & \rho_{(GROSS,TAX)}\sigma_{(GROSS)}\sigma_{(TAX)} \\ \rho_{(NET,GROSS)}\sigma_{(NET)}\sigma_{(GROSS)} & \sigma^2_{NET} & \rho_{(NET,TAX)}\sigma_{(NET)}\sigma_{(TAX)} \\ \rho_{(TAX,GROSS)}\sigma_{(TAX)}\sigma_{(GROSS)} & \rho_{(TAX,NET)}\sigma_{(TAX)}\sigma_{(NET)} & \sigma^2_{TAX} \end{pmatrix}$$
- For each separate variable, calculate the linear combination of the original variable and the random noise as described in paragraph 7 using a parameter δ.

10. The mean vector and the covariance matrix remain the same before and after the perturbation:

$$\boldsymbol{\mu}^{\mathrm{T}}_{\mathrm{Before}} = (6{,}910,\ \ 5{,}343,\ \ 1{,}568),\ \boldsymbol{\mu}^{\mathrm{T}}_{\mathrm{After}} = (6{,}905,\ \ 5{,}339,\ \ 1{,}565),$$

$$\boldsymbol{\sigma}^{\mathrm{T}}_{\mathrm{Before}} = (7{,}181,\ \ 5{,}137,\ \ 2{,}119),\ \boldsymbol{\sigma}^{\mathrm{T}}_{\mathrm{After}} = (7{,}182,\ \ 5{,}139,\ \ 2{,}119),$$

$$\boldsymbol{\rho}_{\mathrm{Before}} = \begin{pmatrix} 1 & 0.9745 & 0.9957 \\ 0.9745 & 1 & 0.9496 \\ 0.9957 & 0.9496 & 1 \end{pmatrix},\ \boldsymbol{\rho}_{\mathrm{After}} = \begin{pmatrix} 1 & 0.9747 & 0.9958 \\ 0.9747 & 1 & 0.9500 \\ 0.9958 & 0.9500 & 1 \end{pmatrix}$$

We used the parameter $\delta = 0.3$ for the linear combination between the original variables and their generated noise. There were only 3 individuals that failed the non-negativity edit E1c based on the variable *tax* and no individuals failed the non-negativity edits for the other income variables *net* and *gross* (edits E1a and E1b). No individuals failed edit E2 for any of the income variables. To correct for the negativity of the variable t*ax*, the value was set to zero and the other variables *gross* and *net* adjusted accordingly to ensure the preservation of the additivity edit E3.

11. Note that in order to generate the multivariate normally distributed noise, we used the Cholesky decomposition (see, e.g., Monahan, 2001), which requires positive definite covariance matrices. In some cases, the covariance matrices had to undergo slight manipulations in order to run the algorithm, which resulted in some small differences in the additivity of the noise variables (up to a value of 10). These were corrected by approportioning the difference between the noise generated for *gross* and the sum of the noise generated for *tax* and *net* relative to their size, and adding on the difference to each noise variable.

III. ROUNDING

12. Rounding to a predefined base is a form of adding noise, although in this case the exact width of the perturbation is known a priori and can be controlled. Therefore, it is likely that micro edits of types E1 and E2 as defined in paragraph 6 will not fail due to the rounding. However, rounding continuous variables separately may cause edit failures of the type defined by E3 since the sum of rounded variables will not necessarily equal their rounded total. Indeed, there are some software applications (and in particular the Tau-Argus Statistical Disclosure Control Software Package developed within the framework of the European Initiative CASC (Salazar-González, Bycroft and Staggemeier, 2005) that have a

controlled rounding option based on sophisticated linear programming which preserves the additivity of the rounded numbers. This method however is biased and in addition, the option is not always available to data suppliers.

13. In our case, where we are dealing with microdata with rather simple edit restrictions, rounding procedures can be relatively easy to implement, similar to the problem of rounding one or two dimensional tables. In this example, we describe a one dimensional random rounding procedure which not only has the property that it is stochastic and unbiased, but it can be carried out in such a way as to preserve the exact overall total (and hence the mean) of the variable being rounded. The algorithm is as follows:

- Let x be the cell value to be rounded and let $Floor(x)$ be the largest multiple k of the base b such that $bk < x$. In addition, define the residual of x according to the rounding base b by $res(x) = x - Floor(x)$. For an unbiased random rounding procedure, x is rounded up to $(Floor(x) + b)$ with probability $res(x)/b$ and rounded down to $Floor(x)$ with probability $(1 - res(x)/b)$. If x is already a multiple of b, it remains unchanged. The expected value of the rounded entry is the original entry. For example, random rounding to base 3 means that all numbers with a residual of 1 are rounded up to 3 with a probability of 1/3 and rounded down to 0 with a probability of 2/3, and all numbers with a residual of 2 are rounded up to 3 with a probability of 2/3 and rounded down to 0 with a probability of 1/3. The expected value of the perturbation in each cell is 0 and the variance is 2.
- The rounding is usually implemented with replacement in the sense that each cell is rounded independently, i.e. a random uniform number u between 0 and 1 is generated for each cell. If $u < res(x)/b$ then the entry is rounded up, otherwise it is rounded down.
- The expectation of the rounding is zero and no bias should remain in the table. However, the realization of this stochastic process on a finite number of cells in a table may lead to overall bias since the sum of the perturbations (i.e., the difference between the original and rounded cell) going down may not necessarily equal the sum of the perturbations going up. In order to preserve the exact total of the variable being rounded, we define a simple algorithm for selecting (without replacement) which cells are rounded up and which cells are rounded down: For those entries having $res(x)$, randomly select $res(x)/b$ of the entries and round upwards, the rest of the entries round downwards. Repeat for all $res(x)$.

14. Rounding as described in paragraph 13 should be carried out within sub-groups in order to benchmark important totals. For example, rounding income in each group defined by age and sex will ensure that the total income in that group will remain unchanged. This may, however, distort the overall total across the whole file, although users are generally more interested in smaller sub-groups for analysis and therefore preserving totals for these groups is more important than the overall total. Reshuffling algorithms can be applied for changing the direction of the rounding for some of the records in order to correct the totals. This algorithm will be described in paragraph 15.

15. For our data set of 16,232 individuals that earned an income in the 2000 Israel Income Survey, we randomly round each of the variables *net* and *tax* to base 10 as described in paragraph 13. The method is carried out separately for each of the variables using the algorithm that controls and preserves the overall total. In order to ensure the edit of additivity E3, we calculate the rounded variable *gross* by summing the rounded variables *net* and *tax*. The rounded variable *gross* now has its overall total preserved (since the individual variables *net* and *tax* had their totals preserved), however since it is derived by adding the two rounded variables, this has caused the resulting sum to jump a base on some of the records. We carry out a reshuffling algorithm to correct this as follows:

- Select the records with more than a difference of 10 (in absolute value) between the original variable *gross* and the rounded variable *gross* that was obtained by summing the rounded variables, *net* and *tax*;
- Determine and select which of the variables *net* or *tax* had the most difference from its original value;
- If the summed rounded variable *gross* was jumped to a higher base, drop the selected variable down a base and if the summed rounded variable *gross* was jumped to a lower base, raise the selected variable up a base.

The results of this procedure are presented in Table 1 and include the impact on the overall totals of each of the variables. Note that ensuring that the summed rounded variable gross is within the base has distorted slightly the controlled total. However the distortion is not large, especially when compared to the alternative of no controls in the totals.

Table 1. Results of the random rounding with and without controls and the re-shuffling algorithm on the totals of rounded variables net, tax *and* gross

Variable	True Total	Random Rounded (no controls on totals and no additivity)	Difference	Random Rounded (controls on totals and additivity but not all within the base)	Difference	Random Rounding (controls on totals and additivity and all within the base)	Difference
tax	25,443,623	25,444,410	-787	25,443,630	-7	25,443,710	-87
net	86,724,755	86,725,330	-575	86,724,770	-15	86,724,860	-105
gross (=*net*+*tax*)	112,168,378	112,169,740	-1,362	112,168,400	-22	112,168,570	-192

IV. MICROAGGREGATION

16. Microaggregation is a disclosure control technique for continuous variables. Records are grouped together in small groupings of size *k*. For each individual in the group, the value of the variable is replaced with the average of the values of the group to which the individual belongs. This method can be carried out both on a univariate or multivariate setting where the latter is implemented through sophisticated computer algorithms. In this paper, we focus on the simple univariate case.

17. Replacing values of variables with their average in a small group will not initiate edit failures of the types described in E1 and E2, although there may be problems at the boundaries and the edits may have to be adjusted slightly. Microaggregation preserves the mean (and the overall total) of the income variable but will lead to a decrease in the variance of the mean because of the following reason:

Let *n* be the sample size, *m* the number of groups of size *p*. The variance components are:

$$SST \qquad \sum_{i=1}^{m}\sum_{j=1}^{p}(X_{ij}-\overline{X})^2 \qquad n\text{-1 degrees of freedom}$$

$$SSB \qquad \sum_{i=1}^{m} p(\overline{X}_i-\overline{X})^2 \qquad m\text{-1 degrees of freedom}$$

$$SSW \qquad \sum_{i=1}^{m}\sum_{j=1}^{p}(X_{ij} - \overline{X}_i)^2 \qquad n\text{-}m \text{ degrees of freedom}$$

The total sum of squares *SST* of the income variable X_i (for i=1,…,n) can be broken down into the "within" sum of squares *SSW* which measures the variance of the mean income variable within the groups and the "between" sum of squares *SSB* which measures the variance of the mean income variable between the groups. When implementing microaggregation and replacing values by the average of their group, the variance that is calculated is based on the *SSB* only and not *SST*. In general, there may not be that much difference between *SST* and *SSB* since the size of the groups is small and this results in a very small *SSW*. In order to minimize this macro edit of a decrease in the variance, therefore, we can generate random noise according to the magnitude of the difference between the two variances and add it to the microaggregated variable. Besides raising the variance back to its expected level, this method will also result in extra protection against the risk of disclosure since Winkler (2002) showed that microaggregation (and in particular univariate microaggregation) can be "unpicked" by intruders using elementary software.

18. We demonstrate this algorithm of adding random noise to a microaggregated variable for the 15,708 individuals that paid a tax from among the 16,232 individuals that earned an income in the 2000 Israel Income Survey. We define small groupings of size 5 where the last grouping may contain more or less than 5 units. We define the groupings within the quintiles as defined in paragraph 6 in order to ensure that micro edits of types E1 and E2 will not begin to fail as a result of adding random noise. In each small group, the value of the variable *tax* is replaced by the average of the group. To generate random noise for each quintile, we calculate the difference between the two variances *SST* and *SSB* and generate the random normal distributed noise with a mean of zero and a variance equal to the difference. Table 2 presents the standard deviations for the mean of the variable *tax* at the different stages of the microaggregation/ additive random noise process. Note that 8 individuals failed edit E2 with a negative value for the perturbed variable *tax*. These individuals had their perturbed value changed to zero.

Table 2. Standard deviation at different stages of microaggregation and random noise for variable tax

	Quintile 1	Quintile 2	Quintile 3	Quintile 4	Quintile 5	Total
Standard deviation of *tax*	79	149	253	555	2,998	2,119
Standard deviation of microaggregated *tax*	61	122	220	502	2,864	2,082
Standard deviation for generating random noise	50	86	125	236	835	394
Standard deviation of microaggregated *tax* with random noise	78	149	252	552	2,981	2,126

For instance, the value 50 in the cell defined by "Standard Deviation for generating random noise" and Quintile 1 is obtained by taking the variance of *tax* (79×79) minus the variance of the microaggregated *tax* (61×61).

19. To ensure the edit constraint E3 based on the additivity of the three income variables, note that carrying out the microaggregation on each of the three variables within group *i* will preserve the additivity since the sum of the means of the two variables *net* and *tax* will equal the mean of the total variable *gross*. In order to ensure the correct variance for the means of the variables, we can generate random noise separately for each variable as described in paragraph 18. However generating random noise separately will not result in preserving the additivity and therefore the linear program technique as described in paragraph 8 will have to be applied.

20. Another method which will preserve the additivity edit E3 is to generate multivariate normal noise which a priori preserves the edit constraint as defined in paragraph 9: $\begin{pmatrix} \varepsilon_{(GROSS)} \\ \varepsilon_{(NET)} \\ \varepsilon_{(TAX)} \end{pmatrix} \sim N(\boldsymbol{\mu}', \boldsymbol{\Sigma})$. For each of the variables, we define the linear combination of the group mean μ_i where *i* is the small group. Let *r*(*i*) be the quintile of *i*. The random noise variable is generated within quintiles. For example, the perturbed variable *gross* in group *i* belonging to quintile *r*(*i*) is equal to: $gross_i' = d_1 \times \mu_i + d_2 \times \varepsilon_{r(i)}$ where $d_1 = \sqrt{(1-\delta^2)}$ and $d_2 = \sqrt{\delta^2}$ as defined in paragraph 7. Since the random multivariate noise itself maintains the additivity property, the additivity will hold when combining the random noise with the group means for each of the three income variables. However, this algorithm will not completely return the original level of the true variance since from paragraph 7:
$Var(gross_i') = (1-\delta^2)Var(\mu_i) + \delta^2 Var(gross_i) = Var(\mu_i) + \delta^2[Var(gross_i) - Var(\mu_i)]$. The last term is the "within" variance and therefore the only way to get back the full covariance structure is to define $\delta = 1$. This however is the definition of synthetic data, which is beyond scope of this paper. By increasing δ slightly we can gain back most of the original variance, although if δ is too high then micro edits of types E1 and E2 will likely begin to fail.

21. We compare these two methods of preserving additivity and ensuring correct variance estimation as described in paragraphs 19 and 20.

- Adding noise separately to each variable, *gross*, *net* and *tax* resulted in correcting the variance but large discrepancies occurred between the sum of variables *net* and *tax* and the total variable *gross*. In this process, 9 records failed edit E1b with a negative perturbed value for *tax*. These values were changed to zero. Table 3 presents the absolute difference between the perturbed variable *gross* and the sum of the perturbed variables *net* and *tax*.

Table 3. Number of individuals with an absolute difference (Diff) between the perturbed variable gross *and the sum of perturbed variables* net *and* tax *based on microaggregation and additive noise*

Diff	Number of Individuals
Total	16,232
No Difference	641
$1 < \text{Diff} \leq 10$	677
$10 < \text{Diff} \leq 50$	2,859
$50 < \text{Diff} \leq 100$	2,966
$100 < \text{Diff} \leq 500$	6,239
$\text{Diff} > 500$	2,850

In order to correct these differences, the linear program technique as described in paragraph 8 was applied. This resulted in the preservation of the additivity constraint, no additional edit failures and also preserved the original ratio between the adjusted perturbed variable *tax* and the adjusted perturbed variable *net*. Table 4 presents the standard deviation of the means for the three variables *gross*, *tax* and *net* at the different stages of microaggregation, additive noise and the linear programming to preserve the edits.

Table 4. Standard deviation at different stages of microaggregation, additive noise and linear programming to correct additivity

Variable	Standard Deviation Original Variable	Standard Deviation Microaggregated Variable	Standard Deviation Microaggregated Variable with Random Noise	Standard Deviation Microaggregated Variable with Random Noise and Additivity
gross	7,181	7,174	7,174	7,174
tax	2,119	2,082	2,115	2,103
net	5,137	5,114	5,134	5,129

- Adding correlated noise with a slightly higher $\delta = 0.5$ preserves the additivity constraint E3. Some edit failures occurred using this high value for δ : 47 out of the 16,232 records had negative values in one of the variables. These were corrected automatically by setting them to zero and adjusting the additivity of the other variables. Table 5 summarizes the standard deviations of the means of the variables *gross*, *tax* and *net* at the different stages of microaggregation and adding correlated random noise.

Table 5. Standard deviation at different stages of microaggregation and correlated additive noise

Variable	Standard Deviation Original Variable	Standard Deviation Microaggregated Variable	Standard Deviation Microaggregated Variable with Random Noise and Additivity
gross	7,181	7,174	7,171
tax	2,119	2,082	2,091
net	5,137	5,114	5,119

22. Comparing these two methods for improving the microaggregation with respect to micro and the macro edit based on the preservation of the variance, it appears that the first method as described in paragraph 19 achieves these aims with the final variance structure closer to the original variance structure. Both of the methods are similar with respect to the resulting correlation structure between the three income variables.

V. RANK SWAPPING

23. In its simplest version, rank swapping is carried out by sorting the continuous variable and defining groupings of size *k*. In each group, random pairs are selected and their values swapped. If the groupings are small, this method will not likely initiate micro edits to fail. In particular, the concern is for micro edits that are based on the logical consistency between highly correlated variables, such as micro edit E2 relating the level of income to age. Other micro edits, for example, would relate income to education qualifications, etc. This is because the method introduces bias on joint distributions that involve the swapped variable. Macro edits that need to be minimized are based on minimizing the distortions to distributions and the effects on statistical inference. The larger the size of the groupings *k* the more possibilities of micro and macro edit failures, however the size of the groupings also impacts inversely on

the disclosure risk, i.e. the larger the groupings the less disclosure risk. Therefore, a balance must be struck based on the parameter *k*, which minimizes micro and macro edit failures and also manages the disclosure risk to a tolerable risk threshold. Note that in order to preserve the edit of additivity as defined in edit E3, all variables involved in the edit would need to be swapped. Otherwise, adjustments could be carried out as defined in paragraph 8 for preserving the additivity.

24. We demonstrate this method on the 16,232 individuals that earned an income in the 2000 Israel Income Survey based on the income variable *gross*. After sorting the variable, we define groupings of size 10 and of size 20, select random pairs in each group and swap the values of *gross* between each pair. No micro edits failed for either size grouping, and the original means and variances for the univariate variable *gross* are preserved. Next we examine some macro edits based on the distortion to a particular joint distribution defined by cross classifying age groups (14), sex (2) and income groups (22). We examine first a distance metric based on the absolute difference between the original and perturbed cell counts of the distribution. Let x_i be the original cell count and $\hat{x}_i$ the perturbed cell count. Also, let n_r be the number of non-zeros cells in the distribution. The distance metric is defined as: $AD = \sum_{i=1}^{n_r} \frac{|x_i - \hat{x}_i|}{n_r}$. A second measure that can be defined reflects on the impact of the distortion to the distribution when carrying out a regression or ANOVA analysis with respect to the "between" variance, i.e., the impact on the R^2 statistic. For an ANOVA analysis, we define the dependent variable as income *gross* and the independent variables as the cross-classified age groups and sex. The measure is the ratio of the "between" variance based on the perturbed distribution and the "between" variance based on the original distribution, where the between variance is defined as: $BV = \frac{1}{p-1}\sum_{i=1}^{p} n_i(\bar{x}_i - \bar{\bar{x}})^2$ where *p* is the number of cells in age group and sex (28 cells), $\bar{x}_i$ is the mean of income within group *i* and $\bar{\bar{x}}$ is the overall mean of *gross*. Table 6 presents the results of these macro edits.

Table 6. Macro edits for the joint distribution of age group, sex and income variable gross

	Groupings of 10	Groupings of 20
Number and Percent of Cells with Differences	106 (22%)	166 (34%)
AD	0.224	0.388
Ratio of BV	-0.03%	0.58%

25. In general, the larger the groupings the more distortions to distributions, although the effect on statistical inference when the dependent variable is the rank swapped variable is small. In contrast, the effects on a statistical analysis when the independent variables are swapped or perturbed are much more severe.

VI. DISCUSSION

26. In this paper we demonstrate how placing controls in the perturbation processes for continuous variables minimizes micro edit failures that are based on preserving the logical consistency of the records. In addition, we focus also on minimizing macro edits, which are based on preserving the quality and utility of the data for statistical analysis and inference. This paper expands on Shlomo and De Waal, 2005, which discussed SDC methods for perturbing categorical variables based on the generalized method of PRAM (the post randomization method). PRAM also encompasses other perturbative methods such as

record swapping and delete/impute. While this paper mainly discusses aspects of utility, quality and consistency, data suppliers and statistical agencies must also focus on minimizing the disclosure risk. The trade off between managing the disclosure risk and ensuring high data utility must be carefully assessed before developing optimal SDC strategies. By combining SDC methods and developing innovative methodologies for implementation, we can preserve sufficient statistics and benchmark totals, and release statistical outputs with higher degrees of utility at little cost to the risk of disclosure.

27. Based on a given threshold for disclosure risk, the "best" method to protect a microdata set is hard to determine in general. For a particular microdata set the "best" SDC method depends on the intended uses of the data by the users, the willingness of the statistical agency to disseminate this data set, the legal aspects of releasing these data, and on the structure of the data. For instance, homogeneous data require different SDC techniques than heterogeneous data. To some extent the "best" SDC method for a microdata set will always be a subjective choice. However, a prerequisite for making a well-founded choice of SDC method is a solid understanding of a wide range of SDC methods. We hope that this paper helps to improve our understanding of several of such SDC methods.

References

Brand, R. (2002), Microdata Protection Through Noise Addition. In *Inference Control in Statistical Databases* (J. Domingo-Ferrer, ed.), New York: Springer, pp. 97-116.

Elamir, E. and C. Skinner (forthcoming), Record-Level Measures of Disclosure Risk for Survey Microdata, *Journal of Official Statistics.*

Fuller, W. A. (1993), Masking Procedures for Microdata Disclosure Limitation, *Journal of Official Statistics*, 9**,** 383-406.

Kim, J.J. (1986), A Method for Limiting Disclosure in Microdata Based on Random Noise and Transformation. American Statistical Association, *Proceedings of the Section on Survey Research Methods*, pp. 370-374.

Monahan, J.F (2001), *Numerical Methods of Statistics*. Cambridge: Cambridge University Press.

Rinott, Y. and N. Shlomo (forthcoming) A Smoothing Model for Sample Disclosure Risk Estimation. *Volume in memory of Yehuda Vardi in the IMS Lecture Notes Monograph Series*, Institute of Mathematical Statistics.

Salazar-González, J.-J., Bycroft, C. and A. T. Staggemeier (2005), Controlled Rounding Implementation. *Joint UNECE/Eurostat Work Session on Statistical Data Confidentiality*, Geneva, November 2005.

Shlomo, N. and T. De Waal (2005), Preserving Edits When Perturbing Microdata for Statistical Disclosure Control. *Statistical Journal of the United Nations ECE 22*, pp. 173-185.

Skinner C. J. and N. Shlomo (2005), Assessing Disclosure Risk in Microdata Using Record Level Measures. *Joint UNECE/Eurostat Work Session on Statistical Data Confidentiality*, Geneva.

Willenborg, L. and T. De Waal (2001), *Elements of Statistical Disclosure Control in Practice*. Lecture Notes in Statistics 155, New York: Springer.

Winkler, W. E. (2002), Single Ranking Micro-aggregation and Re-identification, Statistical Research Division report RR 2002/08, at http://www.census.gov/srd/www/byyear.html.

Yancey, W.E., Winkler, W.E., and R.H. Creecy (2002), Disclosure Risk Assessment in Perturbative Microdata Protection. In: *Inference Control in Statistical Databases* (J. Domingo-Ferrer, ed.), New York: Springer, pp. 135-151.
