

# Tutorial for sdcMicroGUI

Matthias Templ, Bernhard Meindl and Alexander Kowarik

August 2014

International Household Survey Network (IHSN)<sup>1</sup>

---

<sup>1</sup> Acknowledgements: The authors benefited from the support and comments of Shuang Chen (OECD/PARIS21), François Fonteneau (OECD/PARIS21), Olivier Dupriez (World Bank), Geoffrey Greenwell (OECD/PARIS21), Till Zbiranski (OECD/PARIS21) and Marie Anne Cagas (Consultant), as well as from the editorial support of Linda Klinger.

## Table of Contents

1	Overview of sdcMicroGUI .....	4
1.1	How to Use This Tutorial.....	4
2	Getting Started .....	4
2.1	Installation and Updates.....	4
2.2	Main Window .....	5
2.3	Import Data into sdcMicroGUI.....	6
3	Select Identifying Variables.....	7
3.1	Remove Direct Identifiers.....	7
3.2	Select Key Variables.....	8
4	Categorical Variables.....	9
4.1	Disclosure Risk Assessment .....	10
4.2	SDC Methods for Categorical Variables .....	11
4.2.1	Global recoding.....	11
4.2.2	Post-Randomization Method (PRAM).....	14
4.2.3	Local suppression.....	15
5	Continuous Variables.....	16
5.1	Micro-aggregation.....	17
5.2	Adding Noise .....	19
5.3	Shuffling .....	19
6	Export Data and Generate Reports.....	20
7	Reproduce Results Obtained with sdcMicroGUI.....	21
8	Getting Help.....	21
	References.....	22

## List of Figures

Figure 1: On-the-fly preview when importing CSV files .....	6
Figure 2: Main window of the Identifiers tab .....	7
Figure 3: Select variables window .....	8
Figure 4: Generate a strata variable window .....	9
Figure 5: Main window of the Categorical tab .....	9
Figure 6: <i>l</i> -diversity window.....	11
Figure 7: The Frequencies tab in the Global Recoding interface.....	12
Figure 8: Example of recoding age into age groups; original distribution of variable <i>age</i> (continuous) ...	13
Figure 9: Example of recoding age into age groups; variable <i>age</i> recoded into age groups and converted to a factor variable.....	13
Figure 10: The PRAM window.....	14
Figure 11: Optimal local suppression based on <i>k</i> -anonymity .....	15
Figure 12: Local suppression based on risk threshold .....	16
Figure 13: The Continuous tab.....	17
Figure 14: The Micro-aggregation window .....	18
Figure 15: The Shuffling window .....	19
Figure 16: A screenshot of the first page of an internal report .....	20
Figure 17: The View Script window.....	21

# 1 Overview of sdcMicroGUI

The R-based sdcMicroGUI (Kowarik et al., 2013) package allows users to implement common Statistical Disclosure Control (SDC) methods on micro-datasets through easy-to-use, highly interactive, menu-based operations.

The sdcMicroGUI package has the following features:

- **Import/Export:** Data files in various formats, including SAS, SPSS, STATA, CSV files, as well as data stored in R binary format, can easily be imported. After the SDC process is completed, data files can be exported into the same formats.
- **Risks and information loss:** Disclosure risk and information loss measures are automatically calculated and displayed before and after each SDC method is applied, allowing users to instantly assess the impact.
- **Undo:** The *undo* button negates the last command executed on the data. This allows users to try out several methods with different parameters and get instant feedback.
- **Report:** Standardized reports can be generated documenting the SDC process applied to data before release. Internal and external versions of the report are available with different levels of details.
- **Script:** Each procedure implemented through the menu operation is recorded and stored in a script written in R syntax. The script can be saved, modified or re-loaded to reproduce any step or result.
- **Link to sdcMicro:** sdcMicroGUI uses the functionality of the sdcMicro package (Templ et al., 2014b), allowing for high performance and fast computations.

## 1.1 How to Use This Tutorial

This tutorial provides detailed instructions on how to use sdcMicroGUI. Users with no prior knowledge of SDC should first read the guide, *Introduction to Statistical Disclosure Control (SDC)*, to gain a conceptual understanding of the methods and general guidelines on the process; see Templ et al. 2014a.

For more advanced users who are familiar with the native R command line interface, the sdcMicro package provides more extensive applications, and a separate manual is available; see Templ et al. 2014b.

## 2 Getting Started

### 2.1 Installation and Updates

- **Install R.** If R is not yet installed on your computer, go to <http://cran.r-project.org/bin/> and choose your operating system. For Windows users, download the executable file and follow the on-screen instructions. If R is already installed, ensure that it is the latest version.
- **Install sdcMicro and sdcMicroGUI.** Load R and type the following command in R console:

```
>install.packages("sdcMicroGUI")
```

Installation is needed only once. All required packages are automatically installed.

- **Update.** To search for and install available updates, click on the menu item *GUI* → *Check for Updates*, or type in R console:

```
>update.packages()
```

Updates should be installed regularly.

If your organization uses a proxy server to connect to the Internet, automatic access of R is usually restricted, but users can gain the necessary Internet access from within R by calling:

```
>setInternet2 (TRUE)
```

- **Load sdcMicroGUI.** Once the sdcMicroGUI package is installed, type in R console:

```
>require(sdcMicroGUI); sdcGUI()
```

This will load the sdcMicroGUI package into R and display the point-and-click Graphic User Interface (GUI).

## 2.2 Main Window

The main window of sdcMicroGUI consists of the following five menus:

- **GUI** is used to quit, restart or check updates for sdcMicroGUI.
- **Data** is used to import data of various formats into sdcMicroGUI. After the SDC process is completed, use this menu item to export data to various formats and generate reports.
- **Script** is used to import, export and view a script of what has been implemented in sdcMicroGUI.
- **Help** provides links to this tutorial, along with a guideline on SDC (Templ et al. 2014a) and the R help files for sdcMicro (Templ et al. 2014b).
- **Undo** negates the last command executed on the data, allowing users to implement SDC methods in an exploratory manner.

The main window of sdcMicroGUI contains three tabs:

- **Identifiers** contains buttons for selecting/resetting key variables and removing direct identifiers. This tab summarizes the current selection of key variables, displaying number of categories for each categorical key variable, and the minimum, median and maximum for each numerical (or continuous) key variable selected. It also displays summary statistics for any auxiliary variables (e.g., weight, cluster, strata) selected.
- **Categorical** contains SDC methods, disclosure risk measures and information loss measures for categorical variables. The middle column contains buttons for recoding, local suppression and Post-Randomization Method (PRAM). The left column displays disclosure risk measures. The right column displays measures of information loss. These measures are updated each time a SDC method is applied.

- **Continuous** contains SDC methods, disclosure risk measures and information loss measures for continuous variables. The middle column contains options of applying micro-aggregation, adding noise and shuffling. The left column displays risk measures based on record linkage. The right column displays information loss measures. These measures are updated each time a SDC method is applied.

## 2.3 Import Data into sdcMicroGUI

The menu item *Data* offers several options for importing data into sdcMicroGUI. Use *Data* → *Import* for importing datasets in R Dataset, SPSS, SAS or STATA format. Data that are already in the R workspace can be loaded by selecting *Data* → *Choose R-Dataset*.

To import text-delimited CSV files, select *Data* → *Import* → *Import CSV*. A data preview window will pop up showing the first rows of the dataset (Figure 1) and presenting the following options for importing CSV files:

- Check *Header* so the first row of the dataset displays column names
- Check *Fill* to add blanks to rows of unequal lengths
- Check *Strip white* to strip leading and trailing white space from unquoted character fields
- Check *Strings as factors* for converting character vectors to factors
- Check *Blank line skip* to ignore blank lines when reading the file

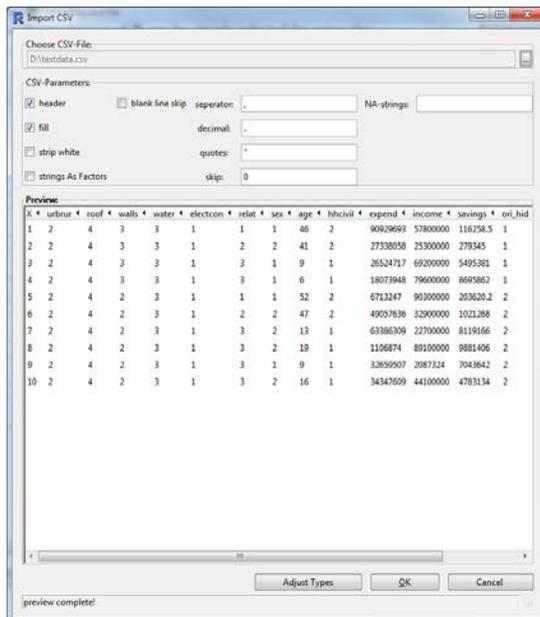


Figure 1: On-the-fly preview when importing CSV files

In the same window, users can also specify the separator between values, decimal operator, quotes, skip and coding of missing values (i.e., NA-strings). Whenever the user changes a field, the data preview

window changes accordingly. Users can also specify the type of each variable (numerical or factor) by clicking the *Adjust Types* button below the preview window.

### 3 Select Identifying Variables

#### 3.1 Remove Direct Identifiers

Once a dataset is imported, the *Remove direct identifiers* button under the Identifiers tab in the main window is enabled (Figure 2). Click on the button to select direct identifiers (e.g., National ID number, name) and remove them from the original data file.

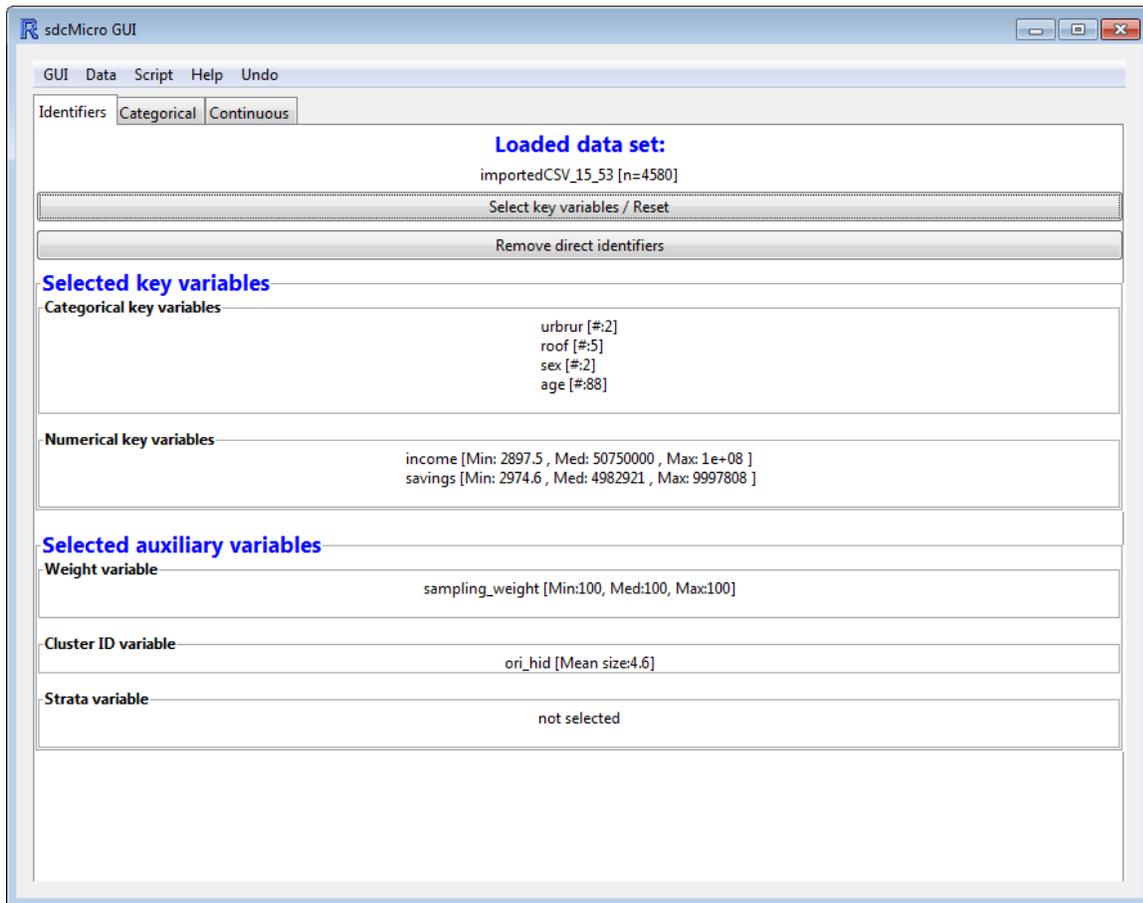


Figure 2: Main window of the Identifiers tab

## 3.2 Select Key Variables

After a dataset is imported, the Select variables window automatically pops up (Figure 3). Users can also click on the *Select key variables/Reset* button under the Identifiers tab (Figure 2) to make or modify the selection at any time.

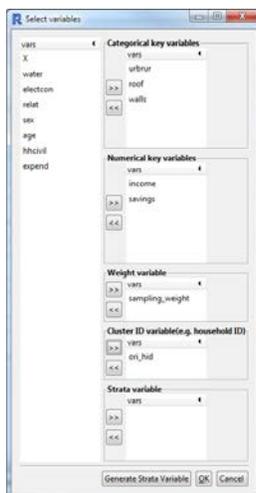


Figure 3: Select variables window

Most SDC methods available in sdcMicroGUI are applied based on the key variables selected in this window<sup>2</sup>, distinguishing categorical (or factor) variables and numerical (or continuous) variables. If a numerical variable is selected as categorical, the Global recoding window (explained in Section 4.2.1) automatically pops up, allowing users to recode it into a categorical variable.

Besides specifying key variables, it is also important to select the weight variable to take into account the survey sampling design.

For hierarchical (or multilevel) micro-datasets, select cluster ID variable (e.g., household ID) to indicate the structure of the data.

In addition, as some SDC methods could be applied to subsets of the data, the user can specify a strata variable. If there is no existing variable for stratification, one can create a new strata variable by clicking the button *Generate Strata Variable* at the bottom of the Select variables window. A new window pops up (Figure 4). The user can select several categorical variables for creating a strata variable. The newly created strata variable will appear in the “vars” list on the left-hand side of the Select variables window; select it as the strata variable.

Note that most of the variable selections are optional; for example, if no numerical key variables are present in the data, users are not required to select any.

---

<sup>2</sup> Except for the PRAM, where all variables can be selected

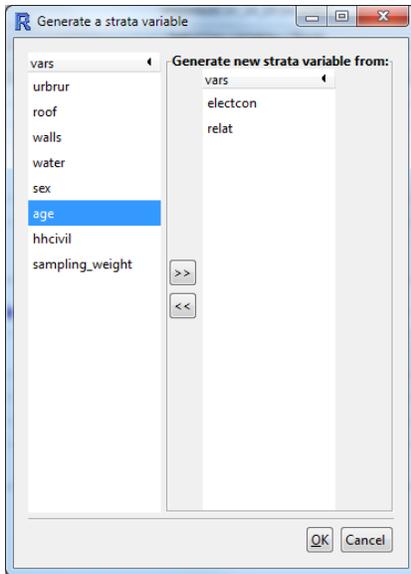


Figure 4: Generate a strata variable window

## 4 Categorical Variables

The Categorical tab in the main window of sdcMicroGUI shows SDC methods applicable to categorical key variables (Figure 5).

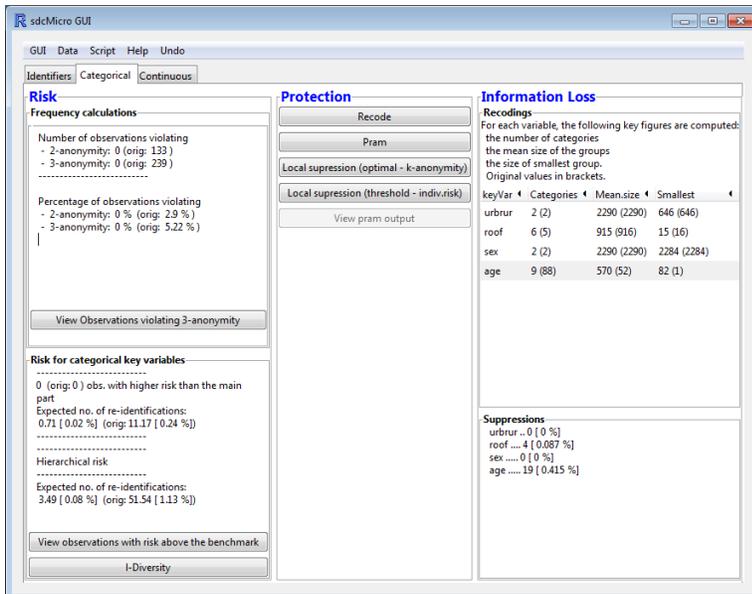


Figure 5: Main window of the Categorical tab

## 4.1 Disclosure Risk Assessment

The left column of the Categorical tab shows disclosure risk measures for categorical variables, and the information is updated whenever a SDC method is applied to the data to illustrate the changes before and after.

The top left column displays the number and percentage of records violating 2-anonymity and 3-anonymity. For example, the output in Figure 5 shows that the original data contain 133 records violating 2-anonymity and 239 records violating 3-anonymity. After applying SDC methods, both numbers were reduced to 0.

To see which records violate 3-anonymity, click on the button *View Observations violating 3-anonymity*. A window pops up listing the sample frequency counts for each record and their variable values.

The bottom left column of the Categorical tab displays global risk measures. The first line indicates the number of records with higher risks than the main part of the dataset<sup>3</sup>. The second indicator is the expected number of re-identifications, estimated by the sum of record-level disclosure risks. If a cluster variable is specified for hierarchical datasets, the expected number of re-identifications at cluster level (e.g., household-level) is indicated<sup>4</sup>.

Observations with high individual risks<sup>5</sup> can be viewed by clicking the button *View observations with risk above the benchmark*. A new window pops up, showing the sample frequency count, individual risk and values of variables for each “unsafe” observation.

The *l-diversity* button on the bottom of the column links to a new window (Figure 6), which allows users to specify sensitive variables and set the parameter *l*. Click *OK* and a separate window pops up, showing the distribution of the distinct *l*-diversity score for each sensitive variable selected. Click on the button *View Observations violating 2-diversity*. A new window opens that lists all records that violate 2-diversity, along with their distinct *l*-diversity scores for each selected sensitive variable, sample frequency counts and values of variables.

---

<sup>3</sup> Records whose individual risk is greater than 0.1 and greater than  $2 \cdot [\text{median}(\mathbf{r}) + 2 \cdot \text{MAD}(\mathbf{r})]$ , where  $\mathbf{r}$  is a vector of record-level risks, and MAD is the median absolute deviation of all record-level risks. See Templ et al. 2014a.

<sup>4</sup> See Templ et al. 2014a for how disclosure risks are defined at cluster level for hierarchical data.

<sup>5</sup> Records whose individual risk is greater than 0.1 and greater than  $2 \cdot [\text{median}(\mathbf{r}) + 2 \cdot \text{MAD}(\mathbf{r})]$ , where  $\mathbf{r}$  is a vector of record-level risks, and MAD is the median absolute deviation of all record-level risks. See Templ et al. 2014a.

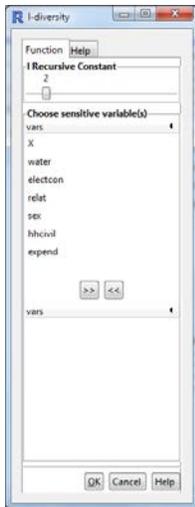


Figure 6: I-diversity window

## 4.2 SDC Methods for Categorical Variables

The middle column of the Categorical tab provides four SDC methods applicable to categorical variables: recoding, PRAM and two different methods for performing local suppression.

### 4.2.1 Global recoding

Click on the button *Recode* in the Categorical tab, and a window for recoding categorical key variables opens. It is possible to recode each categorical key variable separately. If a continuous variable has been selected as categorical, it can be recoded into a categorical variable.

First, select the tab “Frequencies” (Figure 7), which lists number and percentage of observations violating 2-anonymity or 3-anonymity, and shows the distinct combinations of the categorical key variables. For each distinct pattern, a table lists disclosure risks of the records possessing the pattern, sample and population frequency counts<sup>6</sup> ( $f_k$  and  $F_k$ , respectively), as well as the hierarchical risk (if a cluster variable has been selected). The table is sortable, allowing users to identify, for example, unique patterns for which  $f_k = 1$ . Information in this tab is updated every time recoding is applied.

---

<sup>6</sup> The population frequency counts listed here are adjusted sample frequency counts using sampling weights.

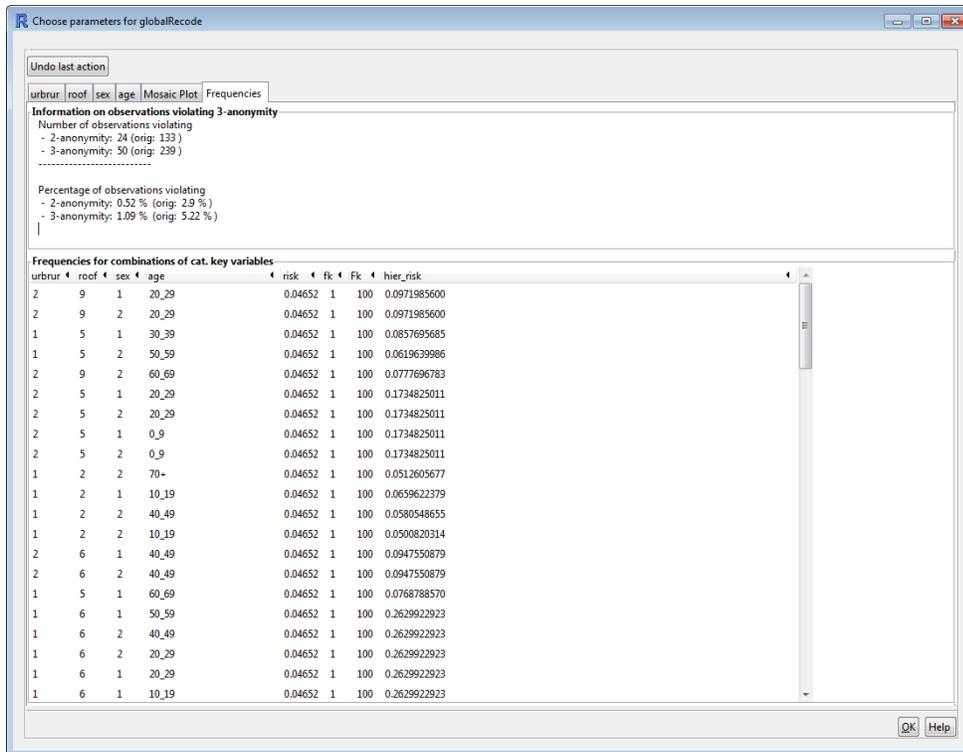


Figure 7: The Frequencies tab in the Global Recoding interface

After examining frequencies and risks of the key categorical variables, click on the tab with the name of the variable to which to apply global recoding.

Users will first specify the variable type: numeric (continuous) or factor (categorical). If numeric is selected, the option *Recode to factor* becomes available, allowing users to recode continuous variables into categorical variables. If factor is selected, the option *Group a factor* becomes available, allowing users to combine categories into one new category. The “Frequencies” box lists the number of categories, values and frequencies for each key variable.

As an example, suppose we have a variable *age*. There are two ways to recode it into a categorical variable indicating age groups (0 to 9, 10 to 19, ...,70 and above). One way is to indicate the variable as numeric, and in the box “recode to factor”, input 0, 9, 19, 29, 39, 49, 59, 69, 119. Click *Recode to factor*, the variable type is automatically changed to *factor*, and the categories of the variable become (0,9], (10, 19], (20, 29]...(70,119]. The option *Group a factor* also becomes available, allowing users to further combine the age groups created. For example, we can select (0, 9] and (10, 19], click *Group selected levels* and name the new level “below 20”.

Alternatively, if we indicate *age* to be a factor variable, the option *Recode to factor* becomes unavailable, while *Group a factor* becomes available. In the “Levels” box, we can select the values 0 through 9, click *Group selected levels* and rename the selected level “0\_9”. Figure 8 and 9 show the window before and after recoding. The bottom right box plots the distribution of the variable.

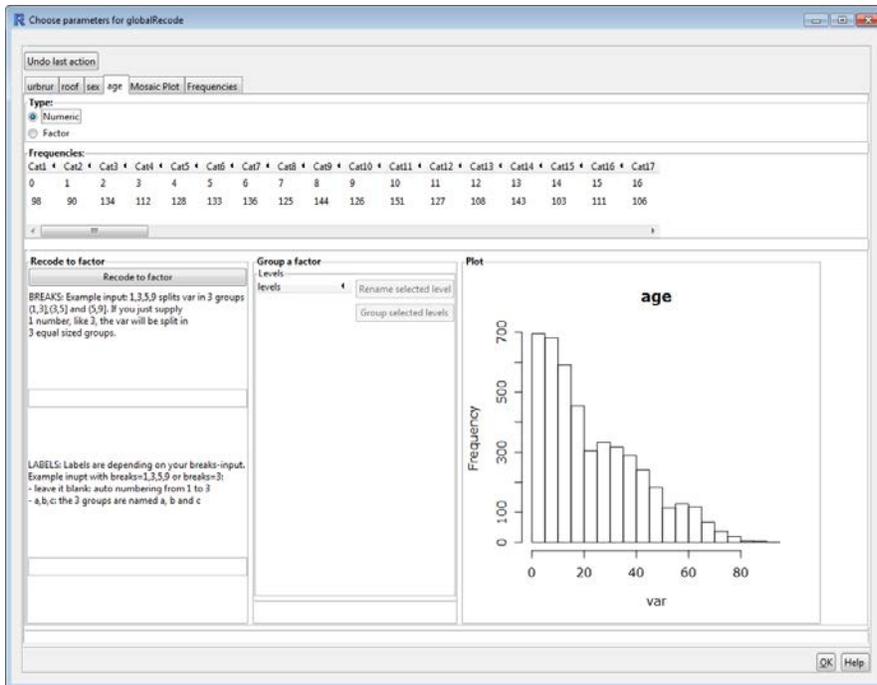


Figure 8: Example of recoding age into age groups; original distribution of variable age (continuous)

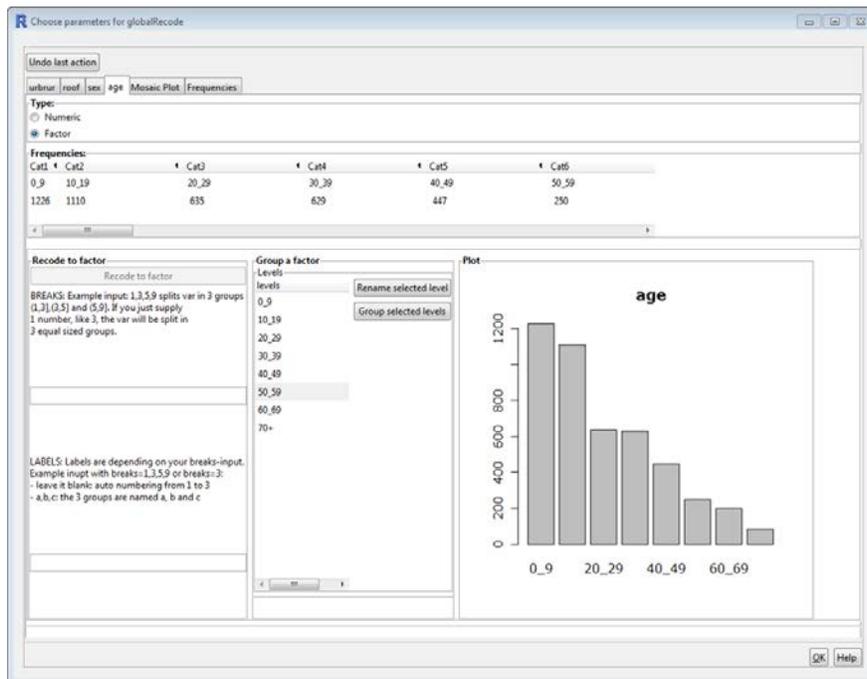


Figure 9: Example of recoding age into age groups; variable age recoded into age groups and converted to a factor variable

In addition, the tab “Mosaic Plot” shows the multivariate distribution of the selected categorical key variables. The plot visualizes the proportion of records possessing each distinct pattern. A pattern with lower proportion indicates lower frequency counts, and thus higher individual risks for the records possessing the pattern.

The impact of recoding on the disclosure risks and information loss of categorical variables can be examined in the main window under the Categorical tab. We can compare the risk measures displayed in the left column. We can also assess the information loss displayed in the top right column, where key variables are listed with number of categories, mean size of categories and the size of the smallest category.

For example, as shown in Figure 5, after we recoded the *age* variable into age groups, the original 88 categories of variable *age* were reduced to 9 and the mean size of categories increased from 52 to 570. Moreover, the size of the smallest age category in the original data is 1 (since *age* was originally a continuous variable), while the size of the smallest age group category is 82 after recoding.

## 4.2.2 Post-Randomization Method (PRAM)

The PRAM (Gouweleeuw et al. 1998) available in *sdcMicroGUI* swaps categories of selected variables based on predefined probabilities<sup>7</sup>.

Click the button *PRAM* under the Categorical tab. A new window pops up, where the user can select the variables to which to apply PRAM (Figure 10). All variables are selectable, because not only can PRAM be applied to key variables, it can also be used to modify sensitive variables violating *l*-diversity. If a strata variable is selected, PRAM is applied on each stratum independently.

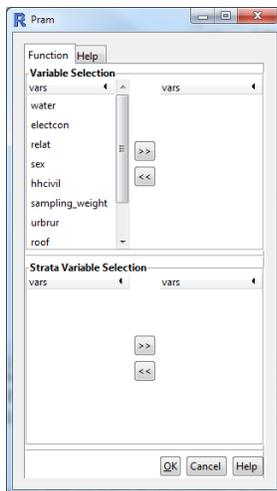


Figure 10: The PRAM window

---

<sup>7</sup> Details on how the default values are set for *sdcMicroGUI* are available in Templ et al. 2014b.

Once the procedure is finished, a results window pops up. The output consists of a three-column table. The first column lists the name of each modified variable; the second and third columns, "nrChanges" and "percChanges", display the number and percentage of values that have been changed for each variable. The results window can also be recalled by clicking the button *View PRAM output* in the main window of the Categorical tab.

### 4.2.3 Local suppression

Even after recoding, some combinations of key variable values may still violate  $k$ -anonymity, or some records may still have relatively high disclosure risks. Further recoding, however, may not be possible because the data utility would be too low. At this stage, local suppression can be applied. Two implementation methods are available in sdcMicroGUI:

The first method aims to achieve  $k$ -anonymity with minimum suppression of values. Click *Local suppression (optimal -  $k$ -anonymity)*, and in the new window (Figure 11), the user can specify the parameter  $k$  for  $k$ -anonymity (typically set to 3 or 4) and rank the selected key variables, with the variable ranked 1 being the most likely to be suppressed. sdcMicroGUI automatically suggests an optimal rank based on the number of unique values for each key variable.

The result after applying local suppression is shown on the top left of the main window under the Categorical tab. Number of key variables violating 2-anonymity or 3-anonymity is automatically updated and compared to the original number.

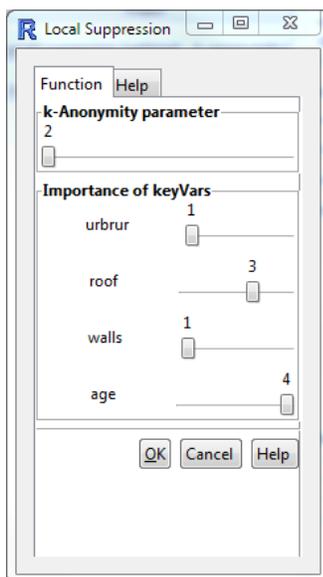


Figure 11: Optimal local suppression based on  $k$ -anonymity

An alternative approach to applying local suppression is to set a record-level risk threshold and suppress values only for records with higher risks than the threshold. Click *Local Suppression (threshold - indiv.risk)*. Adjust the slider "Individual Risk Threshold" to set a record-level risk threshold (Figure 12).

The re-identification rate (estimated by the sum of record-level risks) and number of unsafe records are automatically updated.

Two diagrams are available to illustrate the distribution of record-level risks: *ecdf*, which plots the empirical cumulative density function of the individual risks, and a histogram of the individual risks. In both graphs, a blue line indicates the individual risk threshold specified. Both graphs are updated as the user adjusts the individual risk threshold.

Click *Suppress above threshold* and select a key variable, and all values for that variable will be suppressed (i.e., replaced by missing values) for unsafe records.

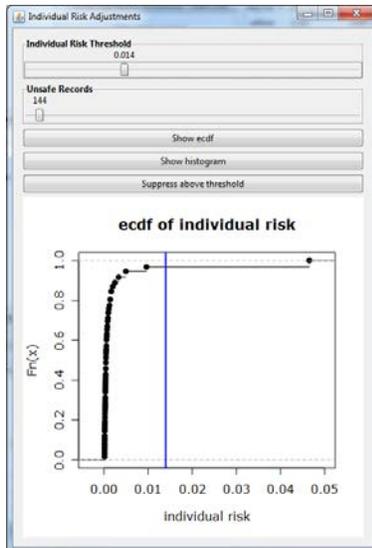


Figure 12: Local suppression based on risk threshold

The extent of information loss after local suppression can be examined in the main window under the Categorical tab; the bottom right of the window lists the number and percentage of values of each key categorical variable suppressed.

## 5 Continuous Variables

The Continuous tab in the main window provides SDC methods for continuous variables and displays disclosure risks and information loss measures (Figure 13). Micro-aggregation, Adding Noise and Shuffling methods can be selected in this window.

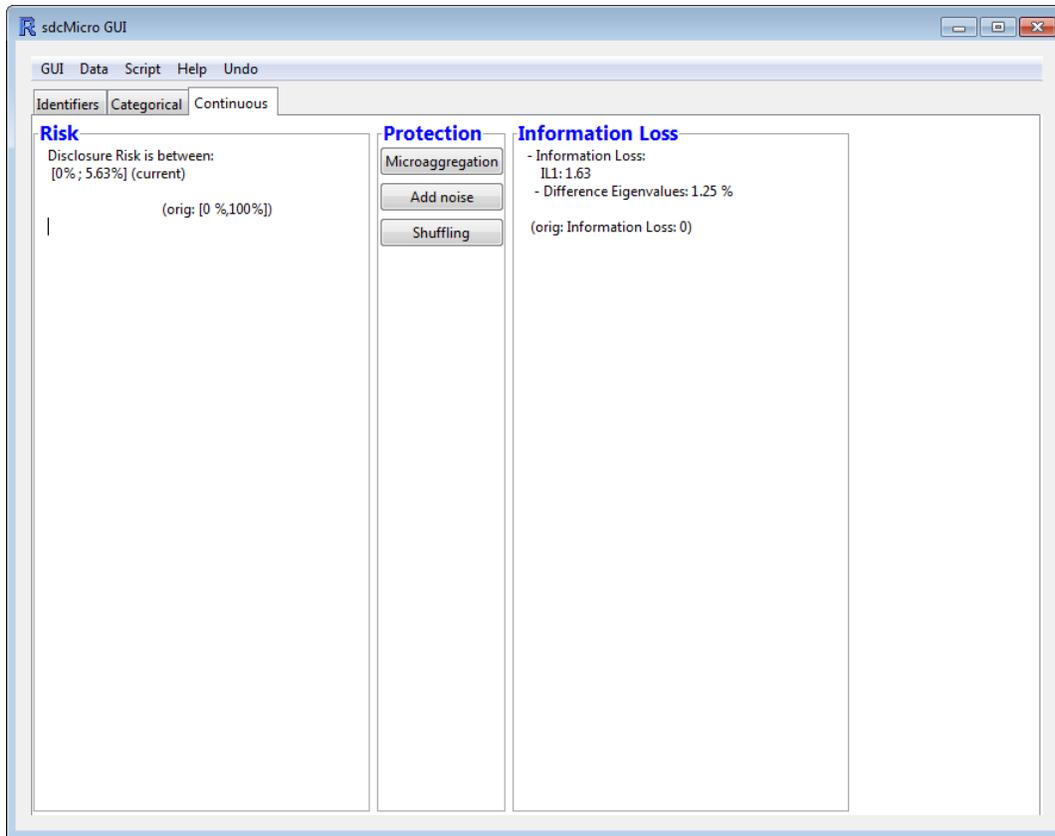


Figure 13: The Continuous tab

After applying any SDC techniques to continuous variables, disclosure risk is displayed in the left column under the Continuous tab. On the left-hand side, two direct measures of information loss, IL1 and differences in eigenvalues (Templ et al. 2014a), are displayed.

The disclosure risk uses the interval disclosure approach (described in Templ et al. 2014a), which constructs an interval around each masked value, and examines whether the original value falls within the interval. The risk is presented as an interval, because the upper bound indicates the proportion of the original values that fall within the interval, assuming the worst-case scenario where every original value that falls within the interval is a correct match with the masked value (i.e. the two values refer to the same respondent). The original data has a disclosure risk between 0% and 100% by default. In the example shown in Figure 13, the interval disclosure risk was reduced to 5.63% after applying SDC methods.

## 5.1 Micro-aggregation

Micro-aggregation partitions records into groups and assigns the arithmetic mean of the group to each variable.

Click on the button *Micro-aggregation*. A new window pops up (Figure 14), where the user selects an aggregation level (i.e., size of the groups), a micro-aggregation method and at least one numerical key variable to which to apply micro-aggregation. The user can also apply micro-aggregation to subsets of the data separately by specifying an additional strata variable.

Five micro-aggregation methods are available in *sdcMicroGUI*:

- **mdav** (e.g., Domingo-Ferrer and Mateo-Sanz, 2002) is the Maximum Distance to Average Vector method that groups records based on classical (Euclidean) distances in a multivariate space
- **rmd** (Templ and Meindl, 2008) groups records based on Robust Mahalanobis Distance
- **pca** (e.g., Templ, 2008) is a projection method that sorts data on the first principal component
- **clustppca** (Templ, 2008) applies the robust counterpart of the *pca* method to clustered data; it is feasible for small- or medium-size datasets
- **influence** (e.g., Domingo-Ferrer et al., 2002) clusters the data and sorts the records by the most influential variable in each cluster

For computational reasons, *mdav* is recommended. It is a highly efficient implementation method, as fast as *pca* but with better performance. For data of medium or small size, *rmd* is a favorable method, as grouping is based on robust distances.

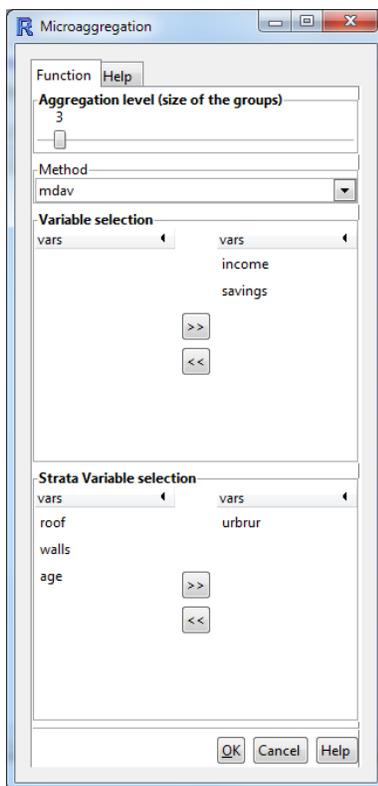


Figure 14: The Micro-aggregation window

## 5.2 Adding Noise

Click *Add Noise*. A new window pops up, where the user specifies whether to add additive or correlated noise (Brand, 2002) using a drop-down menu. Correlated noise is preferred, as it preserves the covariance of the original data. The user must also specify the desired amount of noise, defined as the percentage of co-variance of the continuous key variable in the original data. Select at least one numeric key variable. Click *OK* to add noise to the chosen variable(s).

## 5.3 Shuffling

Shuffling generates simulated values for selected sensitive variables based on the conditional density of sensitive variables given non-sensitive variables. The method then replaces the ranked new values with the ranked original values.

Click *Shuffling*, and the user can select the shuffling method, regression method and co-variance method, respectively, using drop-down menus (Figure 15). Users must also select the response variables (from continuous key variables) and predictor variables. Note that all variables can serve as predictors to generate values for response variables. In *sdcMicroGUI*, selected predictor variables are used in the regression without any interaction terms.

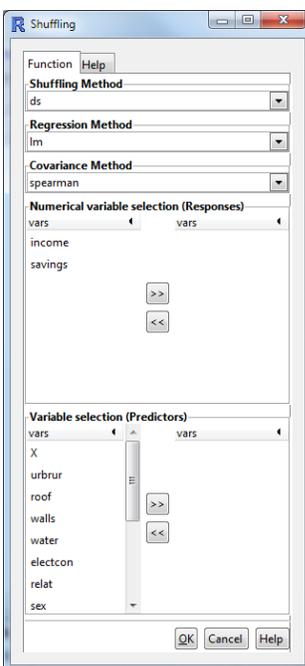


Figure 15: The Shuffling window

The default shuffling method is *ds* (Muralidhar and Sarathy, 2006), but *mvn* (Templ et al., 2014b) and *mlm* (Templ et al., 2014b) may also be selected.

The default regression method is *lm* (linear regression). As a robust alternative, method *MM* (Maronna et al., 2006; Templ et al., 2014b) is also available.

The default co-variance method is *spearman*. Alternative methods include *pearson* and the robust variant *mcd* (Rousseeuw and Van Driessen, 1999).

It is beyond the scope of this tutorial to explain each method in detail. The default settings based on Muralidhar and Sarathy (2006) is recommended. Whether the alternative methods give better results has not been fully tested, but users can explore and compare the different options by examining the resulting information loss and risk measures.

## 6 Export Data and Generate Reports

Once the SDC process is completed, use *Data* → *Export* at the top of the main menu to export the perturbed dataset. The data can be exported as plain text csv files, as well as in formats readable in SAS, SPSS or STATA. In addition, the dataset can be saved directly to the R workspace or by using the R binary format.

Select *Data* → *Generate Report* in the top menu. A new window opens providing the option of producing internal and external reports:

- **Internal reports** include information about the performed actions, disclosure risks, information loss measures and session information on the software versions used. This detailed report is suitable for data producers who hold the original data. Figure 16 shows the first page of the internal report, containing information on selected key variables, SDC methods applied and detailed analysis of risk and utility.
- **External reports** include less information than internal reports. For example, all information on disclosure risks and information loss is suppressed. This report is suitable for external users of the released data.

Reports can be saved as html, pdf or plain text files.

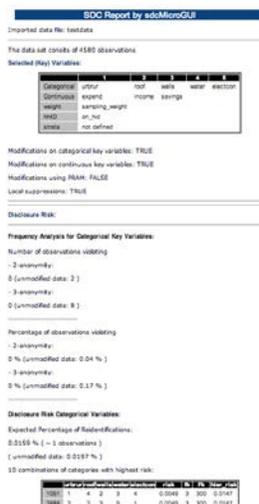


Figure 16: A screenshot of the first page of an internal report

## 7 Reproduce Results Obtained with sdcMicroGUI

Every action the user performs through menu operation is internally recorded and stored in a script. The script can be viewed by selecting *Script* → *View* on the main menu (Figure 17). The scripts can also be exported (*Script* → *Export*) and imported (*Script* → *Import*).

It is possible to edit the scripts. For example, users can remove certain steps from the script or execute some steps only up to a certain point.

This unique feature of sdcMicroGUI is particularly useful for reproducing previous results, resuming work that was started or quickly restarting SDC procedures.

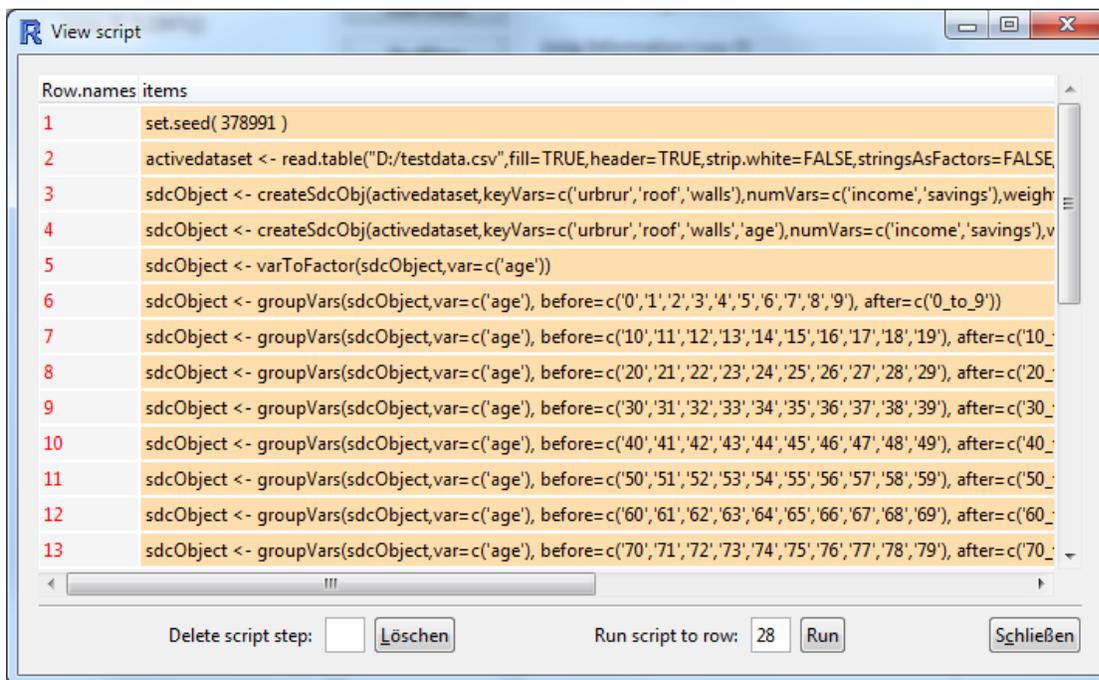


Figure 17: The View Script window

## 8 Getting Help

There are two easy ways to get help with questions about sdcMicroGUI. In the sdcMicroGUI main window, a *Help* menu is available with links to i) this tutorial; ii) the SDC guideline, *Introduction to Statistical Disclosure Control (SDC)* (Templ et al. 2014a) and iii) R sdcMicro help files (Templ et al. 2014b) organized by command. The help files not only describe all possible parameters that can be changed, but also features simple, working examples that can be directly copied into R.

In addition, a *Help* button or tab is available in each SDC method or risk measure window, with information specific to the command.

## References

- R. Brand. Microdata protection through noise addition. In *Inference Control in Statistical Databases*, LNCS 2316, pp. 97-116, 2002. Springer-Verlag Berlin Heidelberg 2002.
- J. Domingo-Ferrer and J.M. Mateo-Sanz. Practical data-oriented microaggregation for statistical disclosure control. *IEEE Trans. on Knowledge and Data Engineering*, 14(1):189-201, 2002.
- J. Domingo-Ferrer, J.M. Mateo-Sanz, A. Oganian and A. Torres. On the security of microaggregation with individual ranking: analytical attacks. *International Journal of Uncertainty, Fuzziness and Knowledge-Based Systems*, 10(5):477-492, 2002.
- J. Gouweleeuw, P. Kooiman, L. Willenborg and P-P. DeWolf. Post randomization for statistical disclosure control: Theory and implementation. *Journal of Official Statistics*, 14(4):463-478, 1998.
- A. Kowarik, M. Templ, B. Meindl and F. Fonteneau. sdcMicroGUI: Graphical user interface for package sdcMicro., 2013. URL <http://CRAN.R-project.org/package=sdcmicrogui>. R package version 1.0.3.
- R. Maronna, D. Martin and V. Yohai. *Robust Statistics: Theory and methods*. Wiley, New York, 2006.
- K. Muralidhar and R. Sarathy. Data shuffling – a new masking approach for numerical data. *Management Science*, 52(2):658-670, 2006.
- P.J. Rousseeuw and K. Van Driessen. A fast algorithm for the minimum covariance determinant estimator. *Technometrics*, 41:212-223, 1999.
- M. Templ. Statistical disclosure control for microdata using the R-package sdcMicro. *Transactions on Data Privacy*, 1(2):67-85, 2008. URL <http://www.tdp.cat/issues/abs.a004a08.php>.
- M. Templ and B. Meindl. Robust statistics meets SDC: New disclosure risk measures for continuous microdata masking. *Privacy in Statistical Databases. Lecture Notes in Computer Science*. Springer, 5262:113-126, 2008. ISBN 978-3-540-87470-6, DOI 10.1007/978-3-540-87471-3\_10.
- M. Templ, B. Meindl, A. Kowarik and S. Chen. *Introduction to Statistical Disclosure Control*. 2014a.
- M. Templ, A. Kowarik and B. Meindl. sdcMicro: Statistical Disclosure Control methods for the generation of public- and scientific-use files. *Manual and Package*. 2014b. URL <http://CRAN.R-project.org/package=sdcmicro>. R package version 4.2.0.