Using the interactive GUI - sdcApp

Bernhard Meindl

2017-01-04

Introduction and Main Features

Package `sdcMicro` contains a shiny app that should help users that are non-experts in R (command-line) to apply disclosure limitation techniques. For this reason, users may upload (micro)data files from different software products into the app and then start to anonymize the dataset by working within the interactive, graphical user interface (GUI). This document will give an overview of the functionalities of the graphical user interface which can be started with `sdcApp()`. The main functionality of the GUI is:

- Uploading microdata

- Modify and analyze microdata
- Creating `sdcMicroObj`-instances
- Perform anonymization techniques on an `sdcMicroObj`-problem instance
- Obtain information on disclosure risks and/or information loss
- Export anonymized data and reports
- Keep reproducibility by being able do download the underlying code from `sdcMicro`

We now describe the features of the interactive graphical user interface in detail. The GUI is separated into 7 main categories, which can be selected from the navigation bar at the top of the screen. Initially, some of these pages will be empty and their content changes once microdata have been uploaded or an `sdcMicroObj` has been generated.

We also want to note that throughout the GUI, questionmark signs are shown. Hovering the mouse over these small icons triggers a pop up window with additional information which will help during the anonymization process.

About/Help

This is the first page that is shown once the graphical user interface has been started using `sdcApp()` after loading package `sdcMicro`. On this page, the user is presented the information on how to open this package vignette which contains extensive information on how to use the GUI. Furthermore, in section `Getting started`, users are advised to either upload microdata or to upload a previously saved problem instance. Clicking on the relevant buttons brings the user automatically to the page from which the desired functionality is available.

In section `Settings`, it is possible to change the default output path. This path is used whenever the user decides to export data from the GUI to the hard disk. The default value is the directory from which the GUI was started (e.g the current working directory). Once a valid path is entered in the text input field, a button labelled `Update the current output path` appears below the input. Pressing this input updates the path. If successful, the current path is shown both as placeholder in the text input as well as in the text above the input field. We note that you can change the path at any time during the anonymization process. Writing files to disk will always use the current path.

From this page, the user can also stop the interface by clicking on a button labelled `Stop the GUI`. If this feature is used, the current [uploaded microdata] after modifications as well as the current problem instance (if it has already been specified) are (invisibly) returned to `R`. So in case the interface was started with `x <- sdcApp()`, `x` then contains a list with two elements named `inputdata` and `sdcObj`. This allows one to continue working in `R`. Finally, users are shown ways on how to contact us for bug reports or any other issues.

We now continue to describe the functionality of the user interface in detail.

Microdata

On this page, the user can either upload data sets stored as files on the hard drive into the GUI or to select data frames that exist in the users workspace before working the graphical user interface was started. This allows to perform common data manipulation steps directly in `R` before continuing to anonymize the dataset using the GUI.

We note that the content of this page changes depending on whether microdata have already been uploaded or not. In the former case, the user can view, modify or reset variables from the uploaded dataset as described in chapter Modify microdata. In the latter case, the user is asked to upload data in the GUI. This is described in chapter Upload microdata below.

Upload microdata

By default, no microdata are loaded into the GUI. Therefore the user has to upload some data in the GUI that can later be anonymized. If no microdata are available, the left-sidebar shows the following options that can be selected by clicking on the appropriate action button. In case the selected data could not be used (eg. the data could not be converted to a `data.frame`), the user is presented with the resulting error message and a button `Try-again`. After clicking this button, another microdata file can be imported.

Use testdata/internal data

This screen allows the user to select data.frames that are available in the users-workspace when starting the user interface. Two test-data sets (`testdata` and `testdata2`, information on which is available from `?testdata`) that are included in `sdcMicro` are always available. Pressing the action button below the drop-down selection input will make the GUI use the selected data frame.

R-Dataset (.rdata)

Here users can opt to upload a file saved in `R` binary format. The users can change the options if character vectors should be automatically converted to factors and if variables that only contain missing-values only should be dropped. By clicking on the `Browse` button the user needs to select a `rdata`-file on disk which he wants to upload. For detailed explanation on the options, please see the chapter on additional options.

SPSS-File (.sav)

Here users can opt to upload a file exported from `SPSS`. The users can change the options if character vectors should be automatically converted to factors and if variables that contain only missing-values ('NA') only should be dropped. By clicking on the `Browse` button the user needs to select a `sav`-file on disk which he wants to upload. For detailed explanation on the options, please see the chapter on additional options.

SAS-File (.sasb7dat)

Here users can opt to upload a file exported from `SAS`. The users can change the options if character vectors should be automatically converted to factors and if variables that contain only missing-values ('NA') only should be dropped. By clicking on the `Browse` button the user needs to select a `sas7bdat`-file on disk which he wants to upload. For detailed explanation on the options, please see the chapter on additional options.

CSV (.csv, .txt)

Here users can opt to upload a text file where variables are separated by some characters. Typically these data would be exported from software such as `Excel`. It is crucial that users indicate if the data file has variable names in the first row and how variables are separated. At this point, users have the option to have character vectors automatically converted to factor or have variables that contain only missing-values ('NA') dropped when the data are read into the GUI. By clicking on the `Browse` button the user needs to select a `txt` or `csv`-file on disk which he wants to upload. For detailed explanation on the options, please see the chapter on additional options.

STATA-File (.dta)

Here users can opt to upload a file exported from `Stata`. The users can change the options if character vectors should be automatically converted to factors and if variables that contain only missing-values ('NA') only should be dropped. By clicking on the `Browse` button the user needs to select a `dta`-file on disk which he wants to upload. For detailed explanation on the options, please see the chapter on additional options.

Additional options

We now describe the choices users can make when uploading data.

Convert character-vectors to factors?

This option is not available when an existing data frame from the current workspace is selected/used. For any other selection, this radio button input has two possible choices, `TRUE` (the default value) and `FALSE`. If `TRUE`, any variables that are read into `R` as character-vectors should be automatically converted to a factor variable. Each distinct value of the variable will be a factor level in the imported dataset. If `FALSE`, no conversion is applied.

Drop variables with only NA-values

This option is not available when an existing data frame from the current workspace is selected/used. For any other selection, this radio button input has two possible choices, `TRUE` (the default value) and `FALSE`. If `TRUE`, any variables in which only `NA`-values are read in are removed from the data set. If this option is set to `FALSE`, these variables (if any) will not be dropped.

First row contains variable names

This option is only available when a text/csv file is imported. This radio button input has two possible choices, `TRUE` (the default value) and `FALSE`. If `TRUE`, the first row of the imported data set will be interpreted as variable names, if `FALSE`, variable names are automatically generated.

Separator (Semicolon, Tab, Colon)

This option is only available when a text/csv file is imported. The radio button input has three possible choices, `Semicolon` (the default value), `Tab` and `Colon` defining the value that is used to separate variables in the input file.

- `Semicolon`: the `;` character is used as separator
- `Tab`: tabulators (`\t`) are used as separators
- `Colon`: the `","` character is used as separator

Select File Input:

This option is not available when an existing data frame from the current workspace should be used. For any other selection, clicking on the `Browse` button allows the user to select a file on the local hard drive. A feature is that only files with the accepted file ending (e.g `.dta` when files from `Stata` `.rdata` when data exported from `R` should be imported) are shown. This reduces the risk, that a unsuitable file can be selected. Once a file has been selected, pushing the `Open` button immediately uploads the file so that the GUI can process it. If the file cannot be read into the system successfully, the user is presented with the resulting error message. If everything works out smoothly, microdata are now available and the left sidemenu changes. The user can now start the anonymization process. For further information, please have a look at the following sections.

Modify microdata

Once data have been uploaded, the content of the *Microdata*-page changes and users can select from a range of possibilities on what to do with the current inputdata. Once data are available, a button `Reset inputdata` is available on top of the sidebar. Clicking this button allows to reset or delete the current input data. However, clicking this button does not immediately reset the problem. Instead, a pop-up window comes up where the user has to confirm to reset the current microdata. This action will be performed, if the user click on the button labelled `Delete current inputdata`. If the user clicks `Dismiss`, the inputdata remains unchanged.

Below this button, a list of action buttons is shown. Clicking on any of these buttons changes the content of the main column. The currently active selection has a different color than the currently inactive buttons. By default, the first entry ("*Display Microdata*") is selected. These entries can be selected by clicking on the desired text or directly on the button. We now continue to describe the features that can be selected.

Display Microdata

This is the default selection, after microdata have been successfully imported or uploaded as described in uploading microdata. This page gives a short overview on the microdata. It shows the name of the imported file as well as the number of observations and the number of variables that are available. Below this information, the user is presented with an interactive table containing the current microdata. The variables can be sorted by clicking on the small arrows next to the variable names on top of the table. Also on the top, there is a dropdown field where users can select how many observations should be displayed on one page. On the bottom of the table the users can find a dynamic pagination field which allows users to jump to a given *"page"* of the current table.

Explore variables

On this page users have the possibility to explore variables from the current microdata. Users have to choose a variable by selecting a variable from the dropdown field with label *"Choose a variable"*. The default value of this input field is the first variable in the dataset. Optionally, a second variable can be selected by choosing a variable from the dropdown field labelled *"Choose a second variable (optional)"* which has the default value of *"None"*. Once the variables have been selected, a graph and additional information is presented below. The specific output depends on the number of variable(s) chosen as well as their type:

One variable selected:

- the selected variable is of type `factor` or `character`:

In this case, a barplot of the factor levels is shown. Below that, a table showing for each factor level the level itself, how often it occurs and the corresponding percentage is shown. Below that, again the number and percentage of missing values is shown.

- the selected variable is of type `integer` or `numeric`:

In this case, a histogram of the selected variable is shown. Below the graph, a table showing main summary statistics (Minimum, Mean, Median, Maximum and 5%-, 25%-, 75%- and 95%-quantiles) are shown. Below this table, the number and percentage of missing values is displayed.

Two variables selected:

- Both variables are of type `integer` or `numeric`:

If both selected variables are continuous (`numeric` or `integer`), a scatterplot of the two variables is displayed. Below that, the correlation coefficient (Pearson) using only pairwise complete observations between the two variables is listed. Below that, two tables are shown. Each table shows the main summary statistics for one of the selected variables. The information included in the tables are (as in the case when only one continuous variable is selected) the Minimum, Mean, Median, Maximum and 5%-, 25%-, 75%- and 95%-quantiles of the variable. Finally, information on the number and percentage of missing values is shown for both variables.

- Both variables are of type `factor` or `character`:

In this case, a mosaicplot of the selected variables is shown as well as a table, containing a cross-tabulation of each levels (or unique values in case of a character input) that shows the number of percentages of each combination of codes given the two selected variables including any combinations with `NA`. Below this table, the number and percentage of missing values is displayed for both selected variables.

- One variables is of type `factor` or `character`, the other variable is of type `integer` or `numeric`:

In this case, a grouped boxplot of the continous variable (type is `integer` or `numeric`) is shown for each level or unique value of the non-continuous variable. Below, for each level of the non-continuous variable, the same summary statistics as already described above of the continuous variable are shown. Finally, the number and percentage of missing values is displayed for both selected variables.

Reset variables

When a microdata set is uploaded, a backup of the unmodified dataset is saved internally. This allows users to reset any modifications in variables in the inputdata file which can be done on this page. To do so, the user needs to select one or more variables from the select field which is by default empty. When all variables that should be reverted are selected, one has to click outside the dropdown field to close the input. Afterwards, an action button labelled `Reset selected variable(s) to their original state` occurs below the input field. Clicking on this button resets the selected variables to their original state. In case you want to remove already selected variables, you can just click on these variable names in the input field and either press the *“backspace”*- or *“delete”*" keys on the keyboard.

In any case, after pressing the action button, the GUI changes to the *“Explore variables”* page at which the first of the selected variable(s) is shown.

Use subset of microdata

On this page it is possible to decrease the size of the input data. This is especially useful if one wants to test different parameter settings and a run on the complete dataset would take long. To reduce the size of the input dataset, the user has to select a method using the drop down field `Select a method to restrict the number of records` and to select a value from the slider `Set 'n' for the selected method`. The following choices are possible on how to reduce the dataset, the range of values that can be chosen from the slider depends on the choice.

- `n-Percent of the data`: `n%` of the data will be randomly chosen. The slider ranges in this case from 1 to 100.
- `the first n-observations`: the data set is decreased by only using the first `n%` records. The slider ranges in this case from 1 to the total number of records.
- `every n-th observation`: This choice allows to create a simple, systematic sample of input data by selecting every `nth` observation. The slider ranges in this case from 1 to at max 500.
- `exactly n randomly drawn observations`: the data set is decreased by taking a random sample of `n%` records. The slider ranges in this case from 1 to the total number of records.

When the desired selections have been applied, pushing the button `Create subset` performs the actual sub-subsampling. After the micro data set has been reduced, the user is taken to the Display microdata page where the reduced dataset can be analyzed.

Convert numeric to factor

This page allows to convert continuous variables (`numeric` or `integer`) into factors. Users can choose from a range of possibilites on how the factor variable should be generated, ranging from automatic conversion to complete manual control.

By default, two input fields are available. On the left hand side there is a dropdown select field termed `Choose numeric variables` where the user can select from the list of numeric variables in the input data set. Next to this element, there is are two radio buttons labelled `Use custom breaks?` with two options, `no` (the default) and `yes`. If the `no` is selected in this input and at least one numeric variable has been clicked from the select input `Choose numeric variable(s)`, a button termed `Convert to factor(s)` occurs. Pressing this button converts all of the selected numerical variables into factors. Afterwards, the user is taken to the explore variables page where the first of the selected variable(s) is shown.

If, however, the radio buttons `Use custom breaks?` are set to `yes`, the layout of the page changes. In this case, the user is able to adjust the way the factor variable should be generated and additional UI elements appear. The first visible change is, that only one numeric variable can now be selected. Now, a list of radio buttons labelled `Choose a numeric variable` is shown where all available variables are printed below each other. Selecting a specific variable works by clicking on either the variable name or the radio button itself. The next choice the user has to make is in a select input field termed `Select algorithm` in which `equidistant` (the default), `logEqui`, `equalAmount` or `manual` are possible values.

The remaining user interface is the same for the first three choices of the `Select algorithm` input and is slightly different if `manual` has been selected. In the first case, a numeric input field labelled `Specify number of intervals` is shown while in the second case a text input field labelled `Specify the custom breaks` occurs right next to the select input field where the algorithm can be selected.

If either `equidistant`, `logEqui` or `equalAmount` are selected, the number specified in the numeric input field defines the number of levels the new factor will have. The difference between the methods are:

- `equidistant`: uses breakpoints that generate intervals of equal length. The number of records in each interval might differ.
- `logEqui`: uses breakpoints that generate intervals of equal length based on the log transformation of the data. The number of records in each interval might differ.
- `equalAmount`: uses breakpoints such that each group/interval has the same number of records. The intervals might be of different length.

Selecting `manual` allows the user to set the breakpoints manually. Note: make sure that all values are included in the specified intervals. The syntax in this text

field is the way that the breakpoints (numbers) have to be entered separated by a colon (,). This sequence of numbers will be interpreted as follows: all values greater than the value before a colon and all values smaller or equal than the value after the colon are grouped together. Any non-matched values will be `NA`. As an example, entering `1,3,5,9` would create a factor from a numeric variable by grouping together all values in x greater than 1 and less or equal to 3 into the first group, all values greater than 3 and less or equal to 5 into the second group and all values greater than 5 and less or equal to 9 into a third group. Any values in x less or equal than 1 or greater or equal to 10 would be `NA`. However, `-Inf` and `Inf` may be entered as the first or last value to avoid the generation of `NAs`. Already existing missing values in the numeric input variable will stay `NA` after the recoding.

What is common for all choices of `Select algorithm` is that a button labelled `Convert to factor` appears if the information that has been entered is correct. Also, at the bottom of the page a table with two columns is shown. The first columns shows each of the unique values of the selected variable while the second column shows the number of occurances of this value. On pressing the action button, the selected numeric variable is recoded according the the parameters that have been chosen. After the recode has been done, the current view changes to the explore variable page with the recoded factor variable already being selected.

Convert variables to numeric

In this page, variables of type `factor` and `character` from the input data can be converted to numeric variables. If no such variables are present in the microdata, the user is shown this information and no conversion is possible. Otherwise, the user can select one or more variables from the input field labelled `Choose variable(s)`. Once at least one variable is selected, a button called `Recode to numeric` occurs below the variable selection input. After clicking this button, the conversion is done and the view changes to the explore variable page with the first of the recoded variables being selected.

It should be noted however - as also shown on the GUI - that this feature should be used with care because internally,function `as.numeric()' is used to perform the conversation. Thus, the resulting numeric vector contains the underlying numeric (integer) representation of the input factor, which is often meaningless as it may not correspond to the factor levels.

Modify factor variable

When this option has been selected from the left hand sidebar, it is possible to modify existing factor variables. The most common use case is to combine one

or more levels of the factor. The other use case is to rename single factor levels. In order to proceed, the user must first select an existing factor variable from the select input labelled `Choose factor variable`. Below this box, there is another input termed `Select Levels to recode/combine`. In this input, all the levels of the active factor variable can be choosen by clicking on the are selectable. Already selected levels may be removed again by clicking on them with the mouse and pressing either the *"backspace"*- or *"delete"* keys on the keyboard.

If at least one factor level is selected, a text input called `New label for recoded values` as well as a radio input labelled `Add missing values to new factor level?` appear. By default, the textbox containing the name of the new level is computed by joining the selected factor levels together using the character _ as separator. By clicking into this text field, users can also start to enter a custom name for the new factor level. If only one level has been selected in `Select Levels to recode/combine`, entering a value different than the default value in this input leads to renaming of this specific factor level. The radio button input labelled `Add missing values to new factor level?` is set to `no` by default. If it is changed to yes, any missing (`NA`) values in the factor are added to the new level.

Below these input fields a button labelled `Group factor levels` and a barplot showing the absolute number of the current levels of the factors are shown. Pressing the action button results in updating the factor. In this case, the page refreshes and the plot adjusts the the changes that have been applied.

Create a stratification variable

On this page, the user is able to generate a new variable based on two ore more variables from the current micro data. The reason for this is that several anonymization techiques that are explained here and here can by applied independently to subgroups of the input data that are given by the values of a so called `stratification variable`. This page allows to create such a variable in a convinient way.

The user has to specify at least two variables in the select field labelled `Select variables to generate a stratification variable`. From this field, all variables that are available in the micro data set can possibly be selected. Once two or more variables are selected, two new inputs appear. The first one appears right next to the variable selection field and is called `Specify variable name for stratification variable`. In this field, user can enter a desired name for the new variable. By default, the variable name listed consists of the selected variables chained together using _ as separator. By clicking into the field, the user can enter a customized variable name. If the current value in this text input field is not the name of an already existing variable, a button termed `Create stratification variable` appears below. Clicking on this button adds the new variable to the input data set. The variable is generated as a factor variable in which the values

of the contributing variables are also chained together using _ as the separator. After pressing the button, the page changes to the explore variable page where the newly generated variable is already selected.

Set specific values to NA

In this section, users can set values in some variables to missing (NA). The first step is to choose one of two possible methods by choosing either by Id (the default selection) or by rule in the radio button input labelled How do you want to select the cells to be recoded to missing? on top of the page.

If by Id has been selected, the user can set values in one or more variables for a specific record to missing. He therefore needs to select at least one variable in the input called Select variable to set records to NA. Once at least one variable has been selected, a new input field occurs. In this input called In which ID do you want to suppress values?, a number between 1 and at most the number of records can be specified. This selection refers to the row in which for the selected variables the values will be set to NA. The user can change this index either by clicking on the small arrows at the right hand side of the input field which allow to increment or decrement the current number by one. As an alternative, it is also possible to directly enter a number in the field.

In case by rule has been selected, these choices are slightly different. In the variable selection it is only possible to select one variable. The other difference is that there is no input field where the user can select a number. Instead, there is a dropdown field where the user can select one of the distinct values of the selected variable. The idea is that all records having this values in the selected variable will the set to NA.

The remaining part of this page is identical for both choices of the method. Below these inputs, an interactive table showing the current microdata is shown. This table can be filtered and navigated exactly the same way as already described here. If all selections are valid, a button labelled Set values to NA is shown above this table. Pressing this button sets the correspondig values to NA in the micro data. Afterwards, the page changes and the Exlore variables page is shown. In this page, the (first) selected variable in which values should be set to missing is pre-selected to be analyzed.

Hierarchical data

In this page, users find the functionality to deal with hierarchical data. The idea is as follows. Often data contain clusters, eg. individual within households or students within classes. In this case, it is often the case that some variables of the data set are only relevant on cluster-level while others are relevant on individual level. It is also often the case that one wants to apply different

anonymization strategies for the different *levels* of the data. So the GUI offers a way to deal with this situation in the following way.

The radio buttons labelled `What do you want to do?` allow to choose from `Prepare file for the anonymization of household level variables` (the default) and `Merge an anonymized household level file into the full dataset`. In the former case, the uploaded micro data set can be restricted to those variables relevant for the cluster-level only. Once that has been done, the user may anonymize the household file and can finally export the anonymized file to disk as described here. In the latter case, an already exported, anonymized household-level file can be imported to the GUI and merged with the individual level file. Then the anonymization process can be started by creating a problem instance using individual-level variables as keys. Finally, the user is able to export an anonymized file that is safe on both levels. We now describe both possibilities in this section:

Prepare file for the anonymization of household level variables

In case the goal is to prepare a household-level file, the user first needs to select an identifier for the households or clusters. This select input termed `Select the household id variable` is initially empty. Once a variable has been selected from the list of all variables available from the input data, an additional select field called `Please select all variables that refer to households and not to individuals` appears next to it. In this input, one or more variables that are relevant for households only (that means, these variables feature the contain the same values for each household) can be selected. Once at least one additional variable has been chosen, a button labelled `Create household-input data` is shown below. Clicking on this button restricts the current data set to the selected variables and to only one (the first) record for each value of the cluster identificator. Finally, the page refreshes and the number of observations and variables in the updated, household level data set is shown. Additionally, the names of the variables as well as their type are presented in tabular format.

Merge an anonymized household level file into the full dataset

In this case, the goal is to merge an already anonymized household level file to the currently available data file. This procedure is performed in two steps. In the first step, the user needs to click on the `Browse` button to select the anonymized file that should be merged. Once the `Open` button is clicked the file is uploaded immediately. We note here that it is only possible to upload data that have been exported as `.rdata`-files as described here. After pressing the button, the file will be uploaded to the system. If an error occured (e.g the selected data file does not contain a data frame or if the data set does not contain any variables that overlap with the current inputdata) an error message is shown and the user may upload a different file from disk.

Once the file is successfully uploaded, the layout of the page changes too. A button labelled `Reset uploaded household data` appears which allows to reset the

household level data and makes it possible to upload yet another file. Additionally, a dropdown field labelled `Select a variable containing household ids` appears. In this input, the user needs to select a variable from the list of variables that are available from both datasets containing the identification variable. The selected variable will finally be used to merge the datasets. Below this selection, a button termed `Merge household- and individual level data` is shown. On clicking this button, the merge is performed. If everything went well, the page refreshes and the number of observations and variables in the updated micro data file shown. Additionally, the names of the variables as well as their type are presented in tabular format.

Anonymize

This page is relevant for creating an sdc problem (of class `sdcMicroObj`) that can then be anonymized within the GUI. If the user navigates to this page and no inputdata have been uploaded, this page shows two options. The user can either click on the button labelled `Upload microdata` or on a button labelled `Upload a previously saved sdc problem`. In the first case the user is taken to the Microdata page where microdata can be uploaded as described here, in the latter case the user is taken to the Undo page where an already exported problem instance can be uploaded.

If microdata are available and no problem has been defined, the user can define a new sdc problem instance. Details on how this can be done are given in chapter Set up a problem. Once a problem instance has been defined, the page layout changes and the user can either view or modify the problem instance as described here or apply anonymization techniques to categorical or continuous variables. Details on how specific methods can be applied are given in chapter Anonymization methods.

Set up a problem

If microdata have already been uploaded, the first step in the anonymization procedure is to create an sdc problem which can be done on this page. The layout of this page is split into two parts. On the left hand side, the user is presented with a table and choices that are required to define a new problem instance. On the right hand side, the user is given the possibility to explore a variable. This is useful for example, to decide which variables should be used as categorical or continuous key variables. For further discussion on the choice of variables, the user should have a look at `?createSdcObj` which is the underlying function that is used to generate a problem. We now explain in detail how to proceed.

On top of the right hand sidebar the user is shown a select input field labelled

`Select variable to show information` in which any of the variables from the current micro data set can be selected. By default the first variable of the data set is chosen. Below this select input a graph depending on the variable type is shown. If the selected variable is either of type `factor`, `character` or `integer` (with less or equal than 10 unique values), a barplot is shown. For variables of type `numeric` or `integer` with more than 10 unique values, a histogram showing the distribution is plotted. Below the plot, the number of unique values including `NA` is shown. Finally, below this information even more information on the selected variable is shown. In case of a continuous variable, the typical main summary statistics (Minimum, Mean, Median, Maximum and 5%-, 25%-, 75%- and 95%-quantiles) are presented while for factor variables (as well as integer variables with not more than 10 unique values), the number of occurences for each possible level is shown.

On the left hand side, an interactive table with a row for each variable available in the microdata and a total of 9 columns is shown. This table allows the user to specify relevant variables for the sdc problem. Also, it shows additional information on each variable. The variables are:

- **Variable.Name**: the variable name
- **Type**: the class of the variable according to `class()`
- **Key**: radio buttons with 3 choices, `no` (the default), `Cat.` and `Cont.`
    - `no`: the variable is not uses as either categorical or continuous key variable
    - `Cat.`: the variable is used as categorical key variable
    - `Cont.`: the variable is used as continuous key variable
- **Pram**: variables that are suitable to be postrandomized
- **Weight**: the variable that contains sampling weights (if any)
- **Cluster.ID**: the variable that identifies clusters (for example households)
- **Remove**: variables that should be excluded when setting up the sdc problem.
- **nrCodes**: the number of unique values in the variable
- **nrNA**: number of missing values in the variable

For columns `Pram`, `Weight`, `Cluster.ID` and `Remove`, checkboxes are present in the table. These checkboxes are by default not selected. The checkboxes can be enabled by clicking on them. While there can at most be one variable selected as weight variable and variable holding cluster ids, multiple variables may be checked in column `Remove` or `Pram`.

Below the table, two slider inputs are shown. The first one, labelled `Parameter "alpha"` is relevant for the frequency calculation given the categorical key variables that contain missing (`NA`) values. For this input, values between `0` and `1` (the default setting) in steps of `0.01` can be selected. We note that leaving the value at `1` leads to the same results as in versions of `sdcMicro` <= 4.7.0. For details on this parameter, please have a look at `?freqCalc`. The second slider,

termed `Parameter "seed"` can take on values between `-250` and `250` in steps of `1` and is the number used to set the seed for the random number generator to ensure reproducability. By default, this value is set to `0`. We note that once the sliders are selected, values can also be increased and decreased by clicking the up and down (or left and right) keys on the keyboard.

Whenever values in either the radio buttons or the checkboxes are changed, it is internally checked if all conditions for a successful generation of a new sdc problem are fulfilled. In the case that some restrictions are violated, either a popup window containing additional information occurs or a red button with the error message is shown. The user can then change the variable settings in the table. Once all checks are passed, a blue button labelled `Setup SDC problem` appears below the two sliders. Clicking on this button creates the sdc problem. Finally, the page refreshes and the layout changes. In Anonymization Methods, these changes are further explained.

Anonymization Methods

Once an sdc problem has been defined as described above, the layout of the page changes. It now features a left sidebar and the main content is shown at the right side of the screen.

In the left sidebar, users can choose which kind of anonymization options they want to apply. On top of the sidebar, a button labelled `Delete SDC problem` is shown. Clicking this button allows to reset the current sdc problem. However, clicking this button does not immediately reset the problem but instead, a popup window appears. In this window the user has to confirm that the current problem should be deleted. This action will be performed, if the user clicks on the button labelled `Delete current problem`. If the user clicks `Dismiss`, the sdc problem remains unchanged.

Below this button, several action buttons are shown which are organized in sections `Reset the Problem`, `View/Analyze existing sdcProblem`, `Anonymize categorical variables` and `Anonymize numerical variables`. By default, the first entry `Show Summary` in `View/Analyze existing sdcProblem` is selected which is made clear due to a different color of the button. The content in the center of the screen is dependent on the choices in the left sidebar. We now continue to describe the possible choices for View/Analyze existing sdcProblem, Anonymize categorical variables and Anonymize numerical variables.

In case anything different from `Show Summary` is selected, the layout of the page is changed again. In this case, a sidebar on the right hand side of the page appears in which many useful statistics on the current anonymization process are listed below each other:

- **Important variables**

In the first section, the *“important”* variables in the current sdc problem are listed. These are the categorical key variables, the numerical key variables (if any) as well as the variables defining sampling weights or cluster identification (if any). For the categorical key variables, the number of suppressions due to establishing `k`-anonymity is also listed in this table.

- **Additional parameters**

The second block lists - also in tabular format - the number of records in the data set within the current sdc problem instance and the value for paramters `alpha` and `random seed` that were used when setting up the current problem.

- **Risk (k-anonymity)**

The next table shows the number and percentages of records violating `2`-, `3`- and `5`-anonymity in the current sdc problem. In parenthesis the corresponding numbers are shown for the initial sdc problem without any anonymization procedures applied.

- **Numeric risk**

In the case that variables have been specified as numerical key variables when setting up the sdc problem, another table showing the estimated minimal and maximum risk for numeric key variables is shown for both the original and the (possibly) modified variables. For more information have a look at `?dRisk`.

- **Information loss**

The section on information loss is also only displayed if continuous key variables are available in the current problem instance. If this is the case, the values for utility measures `IL1s` and the `Difference of Eigenvalues` are shown for both the original and (possibly) modified variables. For more information have a look at `?dUtility`.

This sidebar is always updated whenever the sdc problem instance is modified which is the case when any anonymization procedure was applied. In some cases it is also extended, for example when categorical variables have been post randomized as explained in here and here.

View/Analyze existing sdcProblem

This page allows the user to view the current anonymization state. In the Show Summary page, a lot of detailed information about the current problem instance is shown. After applying anonymization techniques, the GUI often changes to this page so that it is easily possible to check what has changed.

Furthermore, it is also possible to explore the variables within the current problem or to modify the problem instance by linking variables to some categorical key variables as described or to create random identification

variables as it is described here.

Show Summary

This page gives an overview of the current sdc problem. The information listed here is dynamic and is updated whenever an operation (or rather, anonymization technique) has been applied to the problem instance. The summary of the problem is divided into the following subsections. However, not all of these sections are present at any time. The content of the possible parts will be explained in this chapter.

- `Summary of dataset and variable selection`

In this section, information about the dimension (number of records, number of variables) of the current data set in the active problem instance is shown. Additionally, the important variables in the sdcMicro are listed. These variables are:

- *Categorical key variables*: the selected categorical key variables
- *Numerical key variables*: the continuous key variables
- *Sampling weights*: variable containing sampling weights
- *Household/cluster Id*: variable holding household or cluster identificators
- *Deleted variables*: variables that have been deleted when setting up the current problem
- *Linked variables*: variables that are linked to categorical key variables

We note that only the first entry (*Categorical key variables*) is always visible. The other entries are only shown when they were specified when the sdc problem was created as described here.

- `Computation time`

This section prints the current time spent on computations. This refers to the time that was actually spent performing anonymization steps as well as setting up the problem instance. The time shown here does however not track the time that was spent in the GUI.

- `Information on categorical key variables`

In this part of the summary, some aggregation statistics on the categorical key variables are printed in tabular format. The table holds `4` columns and features a row for each categorical key variable of the current problem instance. The columns of this table are:

- *keyVar*: the name of the key variable
- *Number of categories*: the current number of categories
- *Mean size*: the mean size of the existing categories
- *Size of smallest*: The number of records in the smallest category

The the last three columns, the same information based on the data set that was used to create the problem instance is shown in parenthesis. We note that `NA` values (missings) are counted as separate categories in this table.

- `Risk measures for categorical variables`

In the section, the expected number and percentage of re-identifications in the population given the current set of categorical key variables taking account possibly specified sampling weights is printed. Furthermore, a robust measure is shown listing the number of observations whose individual risk is larger than the median of the individual risk distribution plus two times its *"Median Absolute Deviation"*, for details have a look at `?mad`.

The same information is also listed for the initial data set that was used to create the current sdc problem.

- `Information on k-anonymity`

In this section, a table showing the number and percentages of observations that violate `k`-anonymity is shown. The table has the following `3` columns:

- *k-anonimity*: shows the parameter `k`
- *Modified data*: the number and percentages of observations violating `k`-anonymity in the current (anonymized) data
- *Original data*: the number and percentages of observations violating `k`-anonymity in the initial data set used to set up the problem instance.

This table changes for example, if categorical key variables are recoded, k-anonymity is established, postrandomization has been applied (which is described here and here) or values based in their individual risk value are suppressed.

- `Postrandomization`

In case, variables have been postrandomized, as described here and here, the final transition matrices as are shown for each variable that has been post-randomized.

At the end of this section, a table with three columns summarizing the postrandomization results is printed. The columns are:

- *variable*: variable name of variable that has been postrandomized
- *nrChanges*: the absolute number of value-changes.
- percChanges*: the percentage of changed values

For each variable that has been postrandomized, a row is added to this table.

- `Compare numVars`

In this section, a table showing important statistics of numerical key variables is printed. However, this section is only shown if at least one variable has been

specified as numerical key variable when setting up the current problem.

In case it is shown, the table has the following `8` columns.

- *Variable*: the name of the numeric key variable
- *Type*: this shows whether the values in the row refer to the current, possibly anonmyized variable (`modified`) or the initial data used to create the sdc problem (`orig`)
- *Min.*: the minimum value of the variable
- *1st Qu.*: the value at the first quantile
- *Median*: the median value
- *Mean*: the arithmetic mean
- *3rd Qu.*: the value at the third quantile
- *Max.*: the maximium value of the variable

This table is updated, whenever a numeric anonymization technique is applied on at least one numeric key variable.

- `Information on risk for numerical key variables`

This part shows a global risk measure based on the numeric key variables for the current and the initial data set. This information is only visible if at least one variable has been specified as numerical key variable when setting up the current problem.

The assumtion is that the re-identification risk based on numerical key variables is initially always between `0%` and `100%`. The more the numeric key variables are changed, the less the upper bound of this risk interval is.

- `Information loss`

The section on information loss (data utility) of numeric key variables is also only visible if numeric key variables are available in the current problem instance. If this is the case, the values of two measures, `IL1s` and the `difference of eigenvalues` are printed for the current, possibly modified numerical key variables as well as the initial data set used when the problem instance was created. For details on the measures, have a look at `?dataUtility`.

- `Anonymization steps`

At the bottom of this page, the anonymization steps that have been applied, are listed. This helps the user to get a quick overview, on what has already been done to protect the data. This section is expecially useful when previously exported problem instances are imported. If no techniques have been applied, this information is also returned.

Explore variables

This view allows users to explore all variables in their current state in the sdc

problem. The functionality is exactly the same as it was already described in Explore variables for the exploration of variables in the originally uploaded micro data set. The only difference being that the analyzed variables are now those currently available in the active problem instance.

Add 'Ghost'-Variables

Here, users can link one or more variables to a specific categorical key variable. For any linked variable, the anonymized dataset will feature the same suppression pattern than the key variable. This is helpful if for example, similar variables exist but it would not make sense to all add of them as categorical key variables.

In order to link a variable to a key variable, one has to select the key variable using a drop-down menu field labelled `Select categorical key variable`. Next to this input there is another 'select input' field where all variables that are not used as either categorical or numerical key variables, weight- or stratification variable can be selected to be linked to the key variable before. In this input field, multiple variables may be selected.

Once at least one variable has been selected for linking, a button labelled `add 'Ghost'-variables` appears at the bottom of the page. Pressing this button adds the link to the current sdc problem and the view refreshes to the Show Summary page. This information is also displayed on top of the page in the section `Important variables and information`.

Create new IDs

In this part of the GUI it is possible create a new random variable. To perform the task, the user needs to specify two inputs. In the first one, termed `Specify name for the new ID variable`, the desired variable name of the new id needs to be entered. The second input is a drop down field, in which either `none` (the default value) or any variable available from the current sdc problem may be selected. In case a variable has been selected in this input, the newly generated variable features identical (but random) numbers for equal values of the selected variable.

When both inputs have been chosen, a button labelled `Add new ID-variable` appears at the bottom of the page. Pressing this button creates the new variable and adds it to the current sdc problem. The view finally updates and the Show Summary page is shown where the dimension of the data set have been updated.

Anonymize categorical variables

If `Anonymize categorical variables` has been selected in `What do you want to do?`

in the left sidebar of the screen, the options Recoding, k-Anonymity, Postrandomization (simple), Postrandomization (expert) and Supress values with high risk are available from the radio button list termed `Choose a Method` and will be described below.

Recoding

This page allows to recode or reduce the level of detail in the selected categorical key variables. The functionality is the same as already described here for recoding of factor variables in the original microsdcdata file. There are two slight differences, though. The first one is that the variables that can be selected in the input field termed `Choose factor variable` are restricted to the categorical key variables that have been chosen when the sdc problem was created, as described here. The other difference is that once recoding is done, the page refreshes and the content in the right sidebar is recalculated. This especially affects the number of observations violating k-anonymity that are shown in the block `Risk (k-anonymity)`.

k-Anonymity

This section allows to generate `k`-anonymity in the categorical key vars or (independently) within subsets of the key variables. This is done by setting specific values in the categorical key variables to `NA`. Thus, for this method the parameter `alpha` that has been specified during the creating of the sdc problem is of great importance. For a discussion on this parameter, the reader is advised to read the help pages for `?freqCalc`. A feature for this algorithm is that users may enter a preference specifying an order in which variables the required suppressions should take place. Furthermore it is possible to apply the method independently on groups defined by a stratification variable. This is also the first choice the user has to make on this page. In the select input field labelled `Do you want to apply the method for each group defined by the selected variable?` it is possible to select a variable from the set of all variables of type `factor`, `integer` or `character` excluding those variables that have been specified as categorical key variables.

The next input field is termed `Do you want to modify importance of key variables for suppression?`. These radio buttons have two possible choices, `No` (the default) and `Yes`. If `No` is selected, the importance of variables is internally calculated in a way that the more unique values a key variable has, the more likely it is that suppressions in this variable will be done. If `Yes` is selected by clicking on the radio button, the number of additional select input fields appear below. These fields are dynamically labelled `Select the importance for key variable "{var}"` where `{var}` is a placeholder for any categorical key variable. In each of the select inputs, a number between `1` and `n` (the number of key

variables) has be be selected. The key variable that has importance `1` will typically have the least additional suppressed cells while the variable where importance equals `n` will very likely have the largest number of introduced missing values.

Typically, all key variables will be used to determine if `k`-anonymity is reached. If the number of key variables is very large, it is sometimes helpful to establish `k`-anonymity within subsets of the available key variables. If the radio buttons labelled `Apply k-anonymity to subsets of key variables?` is set to `No` (the default choice), all key variables will be used to determine `k`-anonymity. In this case, the user needs to specify the required parameter `k` using a slider input termed `Please specify the k-anonymity parameter`. This slider has by default the value `2` and can take values between `2` and `50`.

If the choice for `Apply k-anonymity to subsets of key variables?` is `Yes`, additional elements appear below. Specifically, for values from `1` to the number of key variables, two additional inputs appear next to each other. The first one is a radio button input field labelled `Apply k-anon to all subsets of {n} key variables?` which is by default set to `No`. If it is set to yes, `k`-anonymity will be established in all combinations of the categorical key variables containing `n` variables. The second parameter is a slider input termed `k-Anonymity-parameter for {n} combs`, which allows to set the parameter `k` for this specific combination. For further details on establishing `k`-anonymity in combination of key variables, please have a look at `?kAnon`.

Once all settings have been applied, a button labelled `Establish k-anonymity` is shown on the bottom of the page. Clicking this button starts the process to establish k-anonymity which might take a long time. On the bottom right screen, a progress bar occurs showing that the process is running. Once it is finished, the page refreshes and the right sidebar is updated. Users should especially have a look at the first table, where the number of suppressions within each key variable is shown. Also, the section `Risk (k-anonymity)` is updated.

Postrandomization (simple)

This page offers the possibility to randomize one or more variables based on an invariant probability transition matrix. To apply this method to the current sdc problem, the user has to choose at least one variable from the input field labelled `Select variable(s) for PRAM`. By default, no variable is selected in this field. The user can select input from a set of variables previously declared suitable for postrandomization. PRAM variables have to be declared while setting up the problem instance in the Anonymize tab.

Once at least one variable that should be pramed has been selected, is it also possible to select a variable which will be used for stratification, from the field named `Postrandomize within different groups (stratification)?`. If the default

value of `no stratification` is changed, the post randomization of the selected variables is performed independently for each unique value of the selected variable. In this field, only one variable may be selected. It should be noted, that stratification variables can be created before setting up the sdc problem instance as it was described here.

To create the transition matrix, two parameters (`pd` and `alpha`) need to be provided using slider inputs. `pd` refers to the minimum diagonal values in the (internally) generated transition matrix. The higher the value chosen, the more likely it is that a value stays in the same category and remains unchanged. Parameter `alpha` allows to add some perturbation to the calculated transition matrix. The lower this number is, the less perturbed the matrix will get. By default, the value of `Choose value for 'pd'` will be `0.8` and the value of `Choose value for 'alpha'` will be `0.5`. For further details, have a look at `?pram`.

After selecting at least one PRAM variable, a button labelled `Postrandomize` appears at the bottom of the page. Pressing this button performs the postrandomization. Afterwards, the page refreshes and in the right sidebar a section called `Postrandomization` either appears or is extended. In this part of the sidebar, for each variable that has been postrandomized the number and percentages of value changes are listed.

Postrandomization (expert)

This page offers the possibility to randomize a variable using a freely specified transition matrix. To apply this method to the current sdc problem, the user has to choose one variable from the input field labelled `Select variable for PRAM` in which by default the first possible variable is selected. The user can choose in this input from any variable that has been specified as a possible variable for postrandomization during the initialization of the sdc problem and that has not yet been pramed in the current sdc problem.

After selecting at least one variable, is it possible to select a variable which will be used for stratification. If, in the input field `Postrandomize within different groups (stratification)?`, the default value of `no stratification` is changed, the post randomization of the selected variables is performed independently for each unique value of the selected variable. In this select field, only one variable may be selected. It should be noted, that stratification variables can be created before setting up the sdc problem instance as it was described here.

Below these input fields, an interactive table is shown. This table has to be edited by the user in a way so that it can be used as a transition matrix. For any given row, the numbers specify percentages that the current value (the actual row name) changes to the value specified by the respective colum name. By default, in the diagonal of the table, the values are `100`. This means that the probability that the value does not change is `100%`. The user can change the table in a way that the sum of the values in each row equals `100`. If this is not the case,

a red button appears below the table giving instant feedback that the table needs to be further edited. Values in specific cells may be changed by clicking into the cell and entering a new values.

Once the transition matrix is valid (eg. the values in all rows sum up to `100`), a button labelled `Postrandomize` appears at the bottom of the page. Pressing this button performs the postrandomization. Afterwards, the page refreshes and in the right sidebar a section called `Postrandomization` either appears or is extended. In this part of the sidebar, for each variable that has been postrandomized the number and percentages of value changes are listed.

Supress values with high risks

On this page the user can set values for the most-risky records to `NA` in a categorical key variable. To do so, the user needs to select a categorical key variable from the select input field labelled `Select key variable for suppression`. By default, the first key variable is already selected. The next step is to set an appropriate threshold value which will be used to identify the *"risky"* records. These records are defined as those having an individual re-identification risk larger than the selected threshold. The threshold may be changed by updating the slider input termed `Threshold for individual risk`. The range of this slider starts at `0` and the maximum value depends on the current sdc problem.

Below these input fields, a histogram showing the distribution of the individual risk values is plotted. In the graph, a vertical black line representing the current value of the threshold is also shown. Finally there is a button labelled `Suppress {nr} values with high risk in variable {var}`. The labelling of this button is dynamic. It shows the number of records that would be set to missing in the selected variable for the current choice of the threshold. If this button is pressed, records in the selected variable whose individual risks are above the threshold are set to `NA`. The view finally updates and the Show Summary page is shown where all measures have been recalculated.

Anonymize numerical variables

If `Anonymize numerical variables` has been selected in `What do you want to do?` in the left sidebar of the screen in the Anonymize page, the options Top/bottom coding, Microaggregation, Adding Noise and Rank Swapping become available from the radio button list termed `Choose a Method`. These methods will be described in the subsequent chapters. We note however that only the first choice (Top/bottom coding) is always available. The remaining choices are only visible if numeric key variables are specified when creating the sdc problem as described here.

Top-/Bottom Coding

This page allows to replace values above (*"Top coding"*) or below (*"Bottom coding"*) a threshold with a custom number. This page not only allows to recode numeric key variables, but any numeric variables currently available. The first step is to choose a variable from the select input labelled `Select variable`. By default, the first numeric variable in the current sdc problem instance is selected. Next to this field are radio buttons labelled `Apply top/bottom coding?` in which by default the value `top` is chosen.

Below these input fields, the user is required to enter two numbers in the input fields labelled `Threshold value` and `Replacement Value`. These numbers relate to the threshold (larger than in case of top coding and less than in case of bottom-coding) for the first input and the number that will replace the current values in the selected variable. To help users find suitable thresholds, a boxplot showing the distribution of the currently selected variable is shown below the inputs.

Once all required input - especially the threshold and the replacement values - is set up and found to be valid, additional elements appear between the input fields and the boxplot. The first additional element is a text stating how many of the values would be replaced as well as the corresponding percentage. Below this information, a button labeled `Apply top/bottom coding` appears. Once this button is pressed, the values are replaced according to the current setting and the page updates with the additional elements disappearing again and the boxplot is updated too. Also, the right sidebar is updated. In case the recoded variable was a numeric key variable, the values in sections `Numeric risk` and `Information loss` may change.

Microaggregation

On this page it is possible to apply microaggregation to numeric (key) variables of the current sdc problem. The user has the choice among a total of `12` different methods. For details on the specific methods, the user is referred to the manual of `?microaggregation`.

The layout of this page changes depending on the specific method that is selected. Microaggregation methods can broadly be categorized into two categories, cluster-based and non-cluster based. This is also the first selection the user can make on this page. Using the radio buttons labelled `Use a cluster-based method?`, the choices `no` (default) and `yes` can be selected by clicking on the appropriate button. The choice in this input field changes the possible selections in the select field termed `Select the method` that is shown next to it. If `Use a cluster-based method?` is `no`, the following methods can be selected:

- `mdav`
- `rmd`
- `simple`
- `single`
- `onedims`

- pca
- mcdpca
- pppca

If `Use a cluster-based method?` is `yes`, the following choices are possible:

- influence
- clustpca
- clustmcdpca
- clustpppca

The next choice the user can make is whether the microaggregation should be performed on the entire data set of independently on groups that are defined by the unique values of a stratification variable. By default, the value `no stratification` is pre-selected in the select field labelled `Apply microaggregation in groups (stratification)?`. The possible variables include all non-numeric variables that are available in the sdc problem. We mention again that stratification variables can be created before setting up the sdc problem instance as described here.

Finally, there are two additional input fields that appear for any of the the microaggregation methods. The first one, labeled `Aggregation-level` is a slider input that defines the size of the groups that should be formed. The value of the slider is by default `3` and it ranges from `1` to `15`. The other input is labelled `Select Variables for Microaggregation`. In this input, the numeric variables that should be microaggregated can be selected from the list of the numeric key variables. If it is empty, all variables will be used. A tooltip once the user hovers over this input field also informs that by default all numeric key variables will be microaggregated.

For some specific methods, additional inputs appear below. For non-clusterbased methods, there are two additional inputs labeled `Aggregation statistics` and `Trimming-percentage` shown below the variable selection input. The input called `Aggregation statistics` is a list of radio buttons with the choices `mean` (the default), `median`, `trim` and `onestep`. If `trim` is selected, a trimmed mean using the value from the slider input labeled `Trimming-percentage` is calculated within each group and this value is used to replace individual values. These additional elements appear for methods `simple`, `onedims`, `pca`, `mcdpca` and `pppca`. For method `simple` a third additional element termed `Select variable for sorting` appears. In this drop down list, the user has to select a variable that will be used to sort the data set before computing the required groups. For details, see `?microaggregation`.

In the case that clusterbased methods should be used, the layout is the same for any of the possible methods. The additional element appear again below the variable selection input `Select variables for microaggregation`. Users can select - as described above - values for `AggregationsStatistics` and the relevant

`Trimming-percentage` if `trim` is selected as the aggregation measure. Furthermore, users can select the desired cluster method in a radio buttons input labeled `Clustermethod` where the choices `clara` (the default), `pam`, `kmeans`, `cmeans` and `bclust` are possible. It is also possible to specify if the data should be transformed before computing the clusters. In the radio buttons list labeled `Transformation`, the choices `none` (the default), `log` and `boxcox` are possible. Finally, the desired number of clusters that should be formed needs to be specified. This number can be set in the slider input labeled `Number of clusters`. By default it is set to `3`.

If all options have been set, a button labeled `Perform Microaggregation` is shown at the bottom of the page. Clicking this button performs the microaggregation of the selected variables according to the options that have been set. Since the computation might take a long time, on the bottom right screen a progress bar appears, showing that the process is running. Once it is finished, the page updates and the Show Summary page is shown. On this page, the section `Compare numVars` is either updated or added, and the sections `Information on risk for numerical key variables`, `Information loss` and `Anonymization steps` are updated to display current values and statistics.

Adding Noise

In this section it is possible to perturb numerical key variables by adding stochastic noise. The first option is to select some numerical key variables from the select input labeled `Select variables`. If this input field is left empty (which is the default), noise will be added to all numerical key variables.

Next to this field, users can select the desired algorithm in the select input termed `Select the algorithm`. The choices are:

- `additive` (the default value)
- `correlated2`
- `restr`
- `ROMM`
- `outdect`
- `correlated`

We note that the last method (`correlated`) is only available if at least two numerical key variables are specified in the current problem instance. For details on the methods, please refer to the section `?addNoise` on the main page.

Below these two input fields, a slider input is shown. This input is dynamically labeled depending on the choice of the method. For all methods, however, this slider is used to enter the amount of perturbation which should be used. Since the parametrization for the different methods is different, this slider has different default values and different ranges depending on the choice of the method. Again, we refer to `?addNoise` for further details.

If all options have been set, a button labeled `Add noise` is shown at the bottom of the page. Clicking this button adds noise to the selected variables according to the options that have been set. Since the computation might take some time, on the bottom right screen a progress bar occurs showing that the process is running. Once it is finished, the page updates and the Show Summary page is shown. On this page, the section `Compare numVars` is either updated or added, and the sections `Information on risk for numerical key variables`, `Information loss` and `Anonymization steps` are updated to display current values and statistics.

Rank Swapping

On this page, the user can apply rank swapping to numerical key variables. For a complete description of the parameters, please see the corresponding main page in `sdcMicro`, `?rankSwap`.

A total of `6` inputs can be set. The first input, labeled `Select variables` allows to select numerical key variables for swapping. If this select field is empty (the default), all numerical key variables will be used. The remaining inputs are all slider inputs defining the required parameters for the algorithm as described in `rankSwap`. The sliders `Percentage of lowest values that are grouped together before rank swapping` and `Percentage of largest values that are grouped together before rank swapping` refer to the top- and bottom- percentages that should be grouped together before the method is applied. Both sliders have by default a value of `0` (the minimum) and can take values up to `25`.

The sliders `Subset-mean preservation factor`, `Multivariate preservation factor` and `Rank range as percentage of total sample size.` allow to fine-tune the algorihm. The first slider refers to argument `K0`, the second to argument `R0` and the third slider to argument `P` in `rankSwap()`. The default values of these sliders are equal to the default values for the function itself and can be changed within reasonable ranges. For details on the impact of these parameters, please see `?rankSwap`.

Once all options have been set, a button labeled `Apply rank swapping` appears at the bottom of the page. Clicking this button applies the algorithm on the selected variables according to the options that have been set. Since the computation might take some time, on the bottom right screen a progress bar appears, showing that the process is running. Once the process is complete, the page updates and the Show Summary page is shown. On this page, the section `Compare numVars` is either updated or added, and the sections `Information on risk for numerical key variables`, `Information loss` and `Anonymization steps` are updated to display current values and statistics.

Risk/Utility

On this tab it is possible to find out current values of various risk measures based on either categorical or numerical key variables in the active sdc problem. It is also possible to visualize or tabulate these variables as well as to identify *“risky”* records in anonymized data set.

If no problem instance has been specified, two buttons will appear. Clicking on `Create a SDC problem` changes the view to the Anonymize page, where a new problem instance can be generated. By pushing the button termed `Upload a previously saved problem`, the view is changed to the Undo page, where a previously saved problem instance can be uploaded.

If a problem instance has been defined, this page features a three column layout. The left sidebar features the navigation that is divided into three section labeled Risk measures, Visualizations and Numerical risk measures. Specific measures can be selected by clicking on the action buttons shown in this sidebar. The current selected button is shown in a different color so that it is easy to see which selection is active.

On the right sidebar, two tables are shown. The first one, labeled `Important variables` lists the categorical and numerical key variables. Additionally (if present), also the variables selected to be possibly postrandomized as well as the variables holding sampling weights or cluster ids are shown. The second table labeled `Additional parameters` shows the number of records as well as the choice of parameters `seed` and `alpha` that were used when the current problem was specified.

The main content depends on the current choice in the navgigation menu. In the following chapters, all possible selections are discussed.

Risk measures

In this section, it is possible to view current values of based on the categorical key variables, identify risky observations of compare plots of individual re-identification risks between original and anonymized micro data which is described here. Also, users may calculate suda2 and l-diversity risk measures.

Information of risk

Here, users can either obtain information on various risk measures based on the categorical key variables; identify risky records; or visualize the individual re-identification risks.

To select what information to view, the user needs to either select `Risk measures` (the default value), `Risky observations` or `Plot of risks` from the radio button list labeled `What kind of results do you want to show?`.

Risk Measures

Here, the number and percentages of observations that have a higher indidvidual re-identification risk than the main part of the other records is shown for both the initial as well as the anonymized data. The individual re-identification risk is computed based on the selected categorical key variables, and reflects both the frequencies of the keys in the data and the individual sampling weights. A record is said to have a re-identification risk different from the main part of the data if its personal re-identification risk is either larger than the median + two times the Median Absolute Deviation of the distribution of all individual risks (a robust measure) or if it is deemed large. For this, the setting was choosen to be `0.1` (`10%`).

Also shown is the number (and corresponding percentages) of observations that are expected to be re-identified. This information is shown for the initial dataset as well as the anonymized data set so that comparisons can be easily done. In case a cluster-variable was specified during the setup of the problem instance, the expected number of re-identifications is also shown if the cluster (e.g persons living in households) information is taken into account as well.

Risky observations

This page allows to filter records in the anonymized data set, depending on a threshold for the individual re-identification risk. To do so, the user can select a specific threshold by moving the slider input labeled `Minimum risk for to be shown in the table`. The slider ranges from `0` (the default value) up to the maximum risk-value currently available in the anonymized data set. If the default value is not changed, all observations are marked as *“risky”* because the re-identification risk is by default larger than `0`. Below the slider, the number and percentages of observations with individual re-identification risks larger than the currently specified threshold are shown. Below, a table containing the categorical key variables, the numbers of `fk`, `Fk` and the individual risk itself are shown for the observations that are marked as *“risky”*. Once the value of the threshold is changed, the number of risky observations will decrease.

Plot of risks

On this page, two plots are presented. The first histogram shows the distribution of the individual re-identification risks in the anonymized data set, while the plot below it shows the same information based on the original data set that was used when the problem instance was created.

Suda2 risk measure

On this page, users can apply the SUDA algorithm. This algorithm can be used to search for Minimum Sample Uniques (MSU) in the data given the current set of key variables. The algorithm looks at those records that are unique in the sample (sample uniques), and checks if any of these sample uniques are also

special uniques. Special uniques are defined as records having keys for which also a subset of the selected key variables is unique in the sample. See the help files for more information on SUDA scores.

We note that this algorithm can only be applied if the current problem instance features three or more categorical key variables. If this requirement is not fulfilled, this information is shown to the user. Else, the user needs to choose a value for parameter `disFraction?` which is the sampling fraction for the simple random sampling or the common sampling fraction for stratified sampling used within the algorithm. By default, this value is set to 0.01 and can be changed by modifying the slider labeled `Specify the sampling fraction for the stratified sampling`.

After pressing the button termed `Calculate suda2-scores`, the actual computation is performed. Once computation is complete, the layout of the page changes. On top of the page a button labeled `Reset to choose a different sampling fraction parameter`. Pressing this button resets the results and allows to recompute the suda2 scores using a different value for parameter `disFraction`. Below this button, two tables are shown. The first table summarizes the suda2 scores that have been obtained. It shows for `0` and 8 intervals the number of records having suda2 scores of this value or within a specific interval. The second table shows for each categorical key variable how much of the total risk is contributed to by each of the variables. This amount is shown in the second column (`contribution`) while the corresponding key variable is listed in column `variable`.

l-Diversity risk measure

Here you can compute the `l`-diversity of sensitive variables. A dataset satisfies `l`-diversity if for every combination of the categorical key variables there are at least `l` different values for each of the sensitive variables. The statistics refer to the value of `l` for each record. To calculate this risk measure, the user needs to first select at least one sensitive variable. This can be done in the input field `Select one or more sensitive variables` where all variables except for the categorical key variables can be selected. The other choice is to set a value for the *l-diversity constant* which can be done using the slider named `Select a value for the recursive constant`. This constant is used to determine if a record is unsafe. If the calculated value for `l-diversity` for a record (having a specific key) is less than this constant, it is said to violate `l`-diversity.

Once the parameters are set, a button labeled `Calculate l-diversity risk measure` appears below. Pressing this button forces the calculation of the measure using the selected sensible variables and the constant. Once the calculation is finished, the content of the page changes. At the top of the page, a button named `Reset to choose different input parameters` is shown. Pressing this button resets the results and allows to specify other parameters. Below, a table containing for each selected sensible variable the 5-number summary of

the calculated l-diversity measure. Below that, all the records that violate `l`-diversity based on the choice of the recursive constant are displayed in an interactive table. If all records are safe, no table is shown.

Visualizations

In this section it is possible to either compare current key variables in the original and anonymized dataset graphically or in tabular format. It is also possible to view measures of information loss based on recoding of categorical key variables or show the number of observations that violate k-anonymity for arbitrary values of `k`.

Barplot/Mosaicplot

On this page it is possible to graphically compare key variables before and after the anonymization. In the select input labeled `Variable 1`, the first categorical key variable is already pre-selected while the value of the second input field `Variable 2` has the default value of `none`. If only one variable is specified, the users is presented with two graphs below these inputs. First, we see a barplot of the original data as it was when the problem instance was created. Below that, another barplot showing the anonymized variable is shown.

If the value in `Variable 2` is different from `none`, the two graphs change. In this case a mosaicplot of the two selected variables is shown for both the original and the anonymized variables.

Tabulations

In this part of the interface it is possible to compare tabulations of categorical key variables before and after the anonymization. The page is built identically as the Barplot/Mosaicplot page. The only difference is that no graphs but tables are displayed below the input fields where the relevant variables can be selected. Also, the tables are shown next to each other to allow for easier comparison and less scrolling.

Information loss

Recoding categorical key variables by combining levels leads to information loss. In this section it is possible to compare for each key variable, the effects of recoding. Thus, a table containing the following columns for each categorical key variable is shown:

- **keyVar**: the name of a categorical key variable

- **nrCategories.orig**: the number of categories in the original variable
- **nrCategories.mod**: the number of categories in the anonymized variable
- **mean.size.orig**: the mean number of elements in each category in the original variable
- **mean.size.mod**: the mean number of elements in each category in the anonymized variable
- **min.size.orig**: the size of the smallest category in the original variable
- **min.size.mod**: the size of the smallest category in the anonymized variable

The table is interactive and in case of many key variables, it can be sorted by clicking on the small arrow signs that are shown next to the column names.

Obs violating k-Anon

On this page it is possible to find out how many records in the anonymized dataset violate `k`-anonymity for different choices of `k`. There is a slider input labeled `Select value for 'k'` that can take values between `1` and `50`. Dragging the slider with the mouse or changing the value of the slider with the arrow-keys on the keyboard leads to a recalculation of the number and percentage of observations that violate `k`-anonymity for the current choice of `k`. This information is printed on the screen below the slider.

Furthermore, a table listing all the observations in the dataset that violate `k`-anonymity is printed. For these observations, the interactive table contains all categorical key variables. Here you can browse the records that violate k-anonymity for the selected level of `k`. All categorical key variables are shown as well as `risk` (the individual risk), `fk` (the frequency of the particular combination of key variables for each record in the sample) and `Fk` (the estimated frequency of this combination of key variables for each record in the population, taking sampling weights into account) are shown.

Numerical risk measures

This section provides information on important summary statistics of numerical key variables in the original and anonymized data; information on the current disclosure risk; as well as measures on information loss.

Compare summary statistics

In this section the user can compare the distribution of the numerical key variables in the current problem, between the original and the anonymized data. The user can also calculate the available measures given the label of a

categorical key variable. To start, the user needs to select a numerical key variable from the select input field labeled `Choose a numerical key variable`. Only pre-selected numerical key variables are available in this field. Next to this input is another select input field labeled `Optionally choose a categorical variable`. Its default value is `None`. In this input field, users may select one of the categorical key variables. If the default value is not changed, the summary statistics shown in tabular form below are calculated for each level of the specified categorical key variable.

Once these selections have been made, some important values are printed in a section named `Measures` below. These values include the Pearson correlation coefficient using pairwise complete information, the standard deviations as well as the interquartile range (a robust measure being the difference between 3rd and 1st quantile of the data set) of the selected variable in the original and the anonymized data set.

Below this information, two tables are presented. The first refers to the original data and shows the Minimum, Mean, Median, Maximum as well as the 5%-, 25%-, 75%- and 95%-quantiles of the selected numerical key variable while the same information is shown in the table below for the variable in the anonymized data set. In case that a categorical variable has been choosen in `Optionally choose a categorical variable`, these summary statistics are calculated for each level of the selected categorical key variable. We note that since the levels of the categorical key variables might differ beetween original and anonymized data set, it is not possible to show this information in a single table.

Disclosure Risk

On this page, users can check on the estimated disclosure risk for the selected numerical key variables. The measure can be interpreted in the following way. In the original, unmodified data that has been used to create the sdc problem, the risk for the numeric key variables is assumed to be between `0%` and `100%`. The more data anonymization techniques such as microaggregation or adding noise are applied to the data, the less the upper bound of the risk will be. So users can compare the estimated upper bound of the risk for numerical key variables in the anonymized data and compare on how much it has reduced from the initial value of `100%`. We note that the larger the deviations from the original data are, the lower the upper risk bound will be. However, this has of course also an impact on data utility measures that can be assessed from the menu button `Information loss` as described below.

Information loss

Here, users can check on two measures of information loss (comment:please name them). Generally speaking, the more the numerical key variables are

modified (anonymized), the higher the information loss values for both measures. We also note that information loss and disclosure risk for numerical variables are always a trade of which need to be balanced.

Also shown on this page, are the values of the `IL1s` measure (definition provided) as well as the differences of robust eigenvalues of the data before and after the anonymization process.

Export Data

In this tab, the GUI offers the possibiliby to export the current state of the anonymized microdata from the current problem instance to a file in various formats, and to save a report summarizing the anonymization process as an html-file to disk. If no problem instance has been specified, the user is informed of the need to create an sdc problem first. By clicking on the button labeled `Create an SDC-Problem`, the GUI changes to the Anonymize page, where the user can create a problem. As an alternative, the user may upload a previously saved problem instance. By clicking on the button `Upload a previously saved problem`, the user is taken to the Undo page where he may upload an previously saved problem instance.

If, however, a problem instance has been defined, the page features a sidebar on the left hand side of the screen. In this sidebar, the user can click on one of two buttons, `Anonymized Data` (the default) or `Anonymization Report`, by clicking on the desired text or button. The active button is finally colored differently and the content of the main page changed depending on your choice.

Anonymized Data

On this page, the microdata available at present in the active sdc problem instance after the applied anonymization techniques as described here can be saved to disk.

On top of this page, an interactive, sortable and browsable table containing showing the data that will be written to a file are shown. The variables can be sorted by clicking on the small arrows next to the variable names on top of the table. Also on the top, there is a dropdown field where users can select how many observations should be displayed on one page. On the bottom of the table the users can find a dynamic pagination field which allows users to jump to a given *“page”* of the current table.

Below the table, two sets of radio buttons are shown:

Select file-format

Using this input, the desired output format can be specified. The possible choices are `R-Dataset`, `SPSS-File`, `Comma-separated File` and `STATA-File` and can

be selected by clicking on the appropriate text or button. If `Comma-separated File` is chosen, additional controls relevant for the generation of the output file appear below. For this option, three additional radio button inputs are available:

- **First row contains variable names**: allows to specify if variable names should be written to the output file (`TRUE`) which is the default setting or not (`FALSE`)
- **Separator**: allows to specify the separation character, possible choices are `Semicolon (;)` (the default), `Tab` and `Comma (,)`
- **Decimal-Character**: allows to specify the decimal character, possible choices are `Dot (.)` (the default) and `Comma (,)`

Randomize Order of Observations

This set of radio buttons allows to choose if the observations in the dataset should be randomized. The possible choices are `Do not randomize` (the default) and `Perform random swapping of IDs`. In the case of the former, the order of records remains unchanged. If `Perform random swapping of IDs` is chosen, the records of the dataset are randomly changed. In the case where a household/cluster variable was selected when specifying the current sdc problem as described here, two additional options are possible. If `Randomize by cluster/household id` is selected, the values of this identification variable are randomized across the dataset. If the user opts to choose `Randomize within cluster/household`, not only are the values of the household identification variable randomly changed, the order of records within the households/clusters is also permuted.

Below this option, a button labeled `Save the anonymized data` is shown. Clicking this button finally creates a file named `exportedData_sdcMicro_{timestamp}.{filetype}` using `writeSafeFile()` with the specified settings in the destination folder that has been specified in the About page or (by default) in the current working directory if this setting has not been changed.

Anonymization Report

On this page, an anonymization report can be generated and saved to disk. The user can select the type of record that can be generated by choosing from the radio button input. If `internal (detailed)` (the default) is selected, a quite long report is generated while the resulting report if `external (short overview)` was selected just gives a very broad overview about the anonymization process. Once the selection has been done, clicking on the button labelled `Save the report` writes the report to disk. A file `sdcReport_internal_{timestamp}.html` is generated in the destination folder that has been specified in the About page or (by default) in the current working directory if this setting has not been changed.

Change Stata Labels

Only if you have uploaded microdata in `dta`-file format, this action button appears. On this page you have the possibility to edit variable labels in an interactive table. These modifications are internally saved and added to the anonymized data file if and only if you choose to export the file as a `dta`-file again as described here.

Reproducibility

In this tab, users find information to be able to reproduce the anonymization steps in the command line interface.

If no inputdata have been uploaded on the Microdata page, this page shows two buttons. Clicking on the button labelled `Upload microdata` sends the user to the Microdata section of the GUI where microdata may be uploaded. Clicking on the button below labelled `Upload a previously saved problem` navigates to the Undo page where a problem instance that has been saved to disk can be uploaded.

If inputdata or a problem instance are available, on the left hand side of the page a sidebar is shown. In this sidebar, users can make the choices that are described below by clicking on the appropriate buttons. It is possible to `View the current script` (the default), `Import a previously exported sdcProblem` from disk or `Export/Save the current sdcProblem` to disk for later re-import. This option is however only possible if a problem instance has already been successfully specified.

View/Save the current script

On this page, users can view the code that has been applied so far. This code could be run in `sdcMicro` directly with the only limitation being that the file path when uploading microdata files is relative to the `fileInput()`-functionality of shiny which gives no way to return the path of the uploaded file on the local disk. So for full reproducibiliby, users may need to adjust the path listed in the current script.

Above the script output, a button labelled `Save Script to File` is shown. Clicking on this button saves the current script in a file `exportedScript_sdcMicro_{timestamp}.R` in the destination folder that has been specified in the About page or (by default) in the current working directory if this setting has not been changed.

Import a previously saved sdcProblem

On this page it is possible to import a previously saved problem instance to the GUI. Once the user clicks on the `Browse` button, they may locate any previously exported problem instance. The file chooser only allows to upload `.rdata` files to minimize possible mistakes. Once the file has been located and the `Open` button is pressed, the selected file is loaded into the GUI. If the import is successful, the content of the GUI is replaced with the data from the imported file.

If the import was not successful, the user is presented with the resulting error message and a button labeled `Try again!`. After clicking this button, it is possible to upload a different file. If the import of the problem instance works, the GUI changes to the overview of the current sdc problem instance as described here.

Export/Save the current sdcProblem

This option is only shown in the sidebar once an sdc problem instance has been generated as described here. If the button labeled `Save the current problem` is clicked, the entire current problem instance (and including all GUI-relevant data) are saved to a file named `exportedProblem_{timestamp}.rdata` in the destination folder that has been specified in the About page or (by default) in the current working directory if this setting has not been changed.

After the file has been successfully saved, the page refreshes and shows the complete path to most recent saved file at the bottom of the page.

Undo

This page allows the user to undo the last anonymization step. If there is no active sdc problem, the user is presented with two options. The user can either click on the button labeled `Upload microdata` in case no microdata have been uploaded to the GUI, or on the button `Create an SDC problem` in case that micro data are available. In the former case the page is changed to the Microdata page while in the latter case it is changed to the Anonymize page. In both cases the user may also click on the `Browse` button to import a previously saved problem instance. We note that this functionality is always available from this page, independent on the availabilty of inputdata, an sdc problem instance, or possibility to undo anonymization steps.

If a problem instance is available, the page layout changes. In case it is possible to undo an anonymization step, the last anonymization action that has been applied is printed on top of this page. Below there is a button termed `Undo last Step`. Clicking on this button opens a pop up window in which the user has to

confirm that the last anonymization step should be reverted. In case this button was unintentionally pressed, clicking on `Dismiss` closes the popup window and it is possible to continue with the anonymization process.

Below, this button there is another action button labeled `Save current state` which has exactly the same functionality as the button described here. Once the problem is successfully saved to disk, the page refreshes and the path the exported file is shown.