

m and **t**-ARGUS: Software for Statistical Disclosure Control

Anco J. Hundepool and Leon C. R. J. Willenborg, *Statistics
Netherlands*

Abstract

*In recent years, Statistics Netherlands has developed a prototype version of a software package, ARGUS, to protect microdata files against statistical disclosure. The launch of the SDC-project within the 4th framework of the European Union enabled us to make a new start with the development of software for Statistical Disclosure Control (Willenborg, 1996). The prototype has served as a starting point for the development of **m**ARGUS, a software package for the SDC of microdata. This SDC-project, however, also plans to develop **t**-ARGUS, software devoted to the SDC of tabular data. The development of these software packages also benefits from the research of other partners in this project. This paper gives an overview of the development of these software packages and an introduction to the basic ideas behind the implementation of Statistical Disclosure Control at Statistics Netherlands.*

Introduction

The growing demands from researchers, policy makers and others for more and more detailed statistical information leads to a conflict. The statistical offices collect large amounts of data for statistical purposes. The respondents are only willing to provide the statistical offices with the required information if they can be certain that these statistical offices will treat their data with the utmost care. This implies that their confidentiality must be guaranteed. This imposes limitations on the amount of detail in the publications. Research has been carried out to establish pragmatic rules to determine which tables can be regarded safe with respect to the protection of the confidentiality of the respondents. The well-known dominance rule is often used.

On the other hand, statistical databases with individual records (microdata files) are valuable sources for research. To a certain extent the statistical offices are prepared to make these microfiles available to researchers, but only under the provision that the information in these databases is sufficiently protected against disclosure. At Statistics Netherlands a lot of research has been carried out to establish rules to determine whether a specific database is safe enough to make it available to researchers. In the next section, we will give an introduction to this research. Then, we will go into the development of μ -Argus and we will conclude with an overview of τ -Argus.

SDC for Microdata at Statistics Netherlands

Re-identification

The aim of statistical disclosure control (SDC) is to limit the risk that sensitive information of individual respondents can be disclosed from a data set (Willenborg and DeWaal, 1996). In case of a microdata set, i.e., a set of records containing information on individual respondents, such disclosure of sensitive

information about an individual respondent can occur after this respondent has been re-identified; that is, after it has been deduced which record corresponds to this particular individual. So, disclosure control should hamper re-identification of individual respondents.

Re-identification can take place when several values of so-called identifying variables, such as "Place of residence," "Sex," and "Occupation" are taken into consideration. The values of these identifying variables can be assumed to be known to friends and acquaintances of a respondent. When several values of these identifying variables are combined, a respondent may be re-identified. Consider for example the following record obtained from an unknown respondent:

"Place of residence = Urk," "Sex = Female" and "Occupation = Statistician."

Urk is a small fishing village in the Netherlands, in which it is unlikely for many statisticians to live, let alone female statisticians. So, when we find a statistician in Urk, a female one moreover, in the microdata set, then she is probably the only one. When this is indeed the case, anybody who happens to know this rare female statistician in Urk is able to disclose sensitive information from her record if it contains such information.

An important concept in the theory of re-identification is a *key*. A key is a combination of identifying variables. Keys can be applied to re-identify a respondent. Re-identification of a respondent can occur when this respondent is rare in the population with respect to a certain key value, i.e., a combination of values of identifying variables. Hence, rarity of respondents in the population with respect to certain key values should be avoided. When a respondent appears to be rare in the population with respect to a key value, then disclosure control measures should be taken to protect this respondent against re-identification (DeWaal and Willenborg, 1995a).

In practice, however, it is not a good idea to prevent only the occurrence of respondents in the data file who are rare in the population (with respect to a certain key). For this, several reasons can be given. Firstly, there is a practical reason: rarity in the population, in contrast to rarity in the data file, is hard to establish. There is generally no way to determine with certainty whether a person who is rare in the data file (with respect to a certain key) is also rare in the population. Secondly, an intruder may use another key than the key(s) considered by the data protector. For instance, the data protector may consider only keys consisting of at most three variables, while the intruder may use a key consisting of four variables. Therefore, it is better to avoid the occurrence of combinations of scores that are *rare* in the population in the data file instead of avoiding only population-uniques in the data file. To define what is meant by rare, the data protector has to choose a threshold value D_k , for each key value k , where the index k indicates that the threshold value may depend on the key k under consideration. A combination of scores, i.e., a key value, that occurs not more than D_k times in the population is considered *unsafe*; a key value that occurs more than D_k times in the population is considered *safe*. The unsafe combinations must be protected, while the safe ones may be published.

There is a practical problem when applying the above rule that the occurrence (in the data file) of combinations of scores that are rare in the population should be avoided. Namely, it is usually not known how often a particular combination of scores occurs in the population. In many cases, one only has the data file itself available to *estimate* the frequency of a combination of scores in the population. In practice, one therefore uses the estimated frequency of a key value k to determine whether or not this key value is safe or not in the population. When the *estimated* frequency of a key value, i.e., a combination of scores, is larger than the threshold value D_k , then this combination is considered *safe*. When the *estimated* frequency of a key value is less than or equal to the threshold value D_k , then this combination is considered *unsafe*. An example of such a key is "Place of residence," "Sex," and "Occupation."

SDC Techniques

Statistics Netherlands, so far, has used two SDC techniques to protect microdata sets, namely global recoding and local suppression. In case of global recoding, several categories of a variable are collapsed into a single one. In the above example, for instance, we can recode the variable “Occupation.” For instance, the categories “Statistician” and “Mathematician” can be combined into a single category “Statistician or Mathematician.” When the number of female statisticians in Urk plus the number of female mathematicians in Urk is sufficiently high, then the combination “Place of residence = Urk,” “Sex = Female,” and “Occupation = Statistician or Mathematician” is considered safe for release. Note that instead of recoding “Occupation,” one could also recode “Place of residence” for instance.

The concept of MINimum Unsafe Combinations (MINUC) plays an important role in the selection of the variables and the categories for local suppression. A MINUC provides that suppressing any value in the combination yields a safe combination.

It is important to realize that global recoding is applied to the whole data set, not only to the unsafe part of the set. This is done to obtain a uniform categorization of each variable. Suppose, for instance, that we recode “Occupation” in the above way. Suppose furthermore that both the combinations “Place of residence = Amsterdam,” “Sex = Female,” and “Occupation = Statistician,” and “Place of residence = Amsterdam,” “Sex = Female,” and “Occupation = Mathematician” are considered safe. To obtain a uniform categorization of “Occupation” we would, however, not publish these combinations, but only the combination “Place of residence = Amsterdam,” “Sex = Female,” and “Occupation = Statistician or Mathematician.”

When local suppression is applied, one or more values in an unsafe combination are suppressed, i.e., replaced by a missing value. For instance, in the above example we can protect the unsafe combination “Place of residence = Urk,” “Sex = Female” and “Occupation = Statistician” by suppressing the value of “Occupation,” assuming that the number of females in Urk is sufficiently high. The resulting combination is then given by “Place of residence = Urk,” “Sex = Female,” and “Occupation = missing.” Note that instead of suppressing the value of “Occupation,” one could also suppress the value of another variable of the unsafe combination. For instance, when the number of female statisticians in the Netherlands is sufficiently high then one could suppress the value of “Place of residence” instead of the value of “Occupation” in the above example to protect this unsafe combination. A local suppression is only applied to a particular value. When, for instance, the value of “Occupation” is suppressed in a particular record, then this does not imply that the value of “Occupation” has to be suppressed in another record. The freedom that one has in selecting the values that are to be suppressed allows one to minimize the number of local suppressions. More on this subject can be found in De Waal and Willenborg (1995b).

Both global recoding and local suppression lead to a loss of information, because either less detailed information is provided or some information is not given at all. A balance between global recoding and local suppression has to be found in order to make the information loss due to SDC measures as low as possible. It is recommended to start by recoding some variables globally until the number of unsafe combinations that have to be protected by local suppression is sufficiently low. The remaining unsafe combinations have to be protected by suppressing some values.

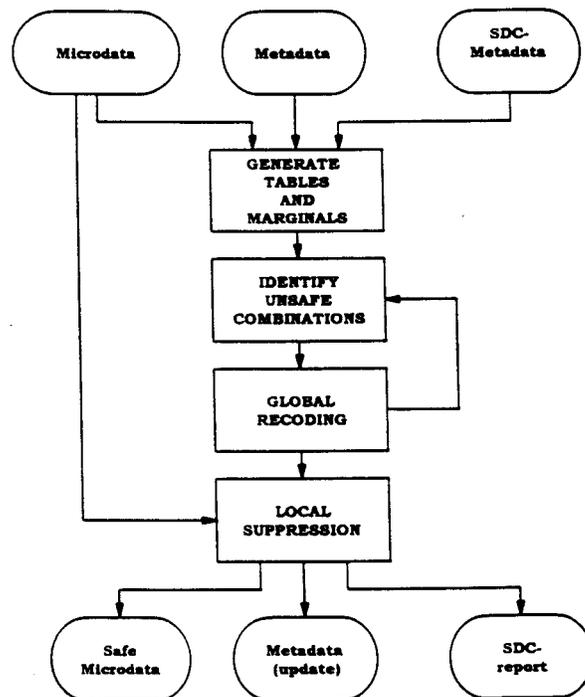
μ -ARGUS allows the user to specify the global recodings interactively. The user is provided by μ -ARGUS with information, helping him to select these global recodings. In case the user is not satisfied with a particular global recoding, it is easy to undo it. After the global recodings have been specified the values that have to be suppressed are determined automatically and optimally, i.e., the number of values that have to be suppressed is minimized. This latter aspect of μ -ARGUS, determining the necessary local suppressions

automatically and optimally, makes it possible to protect a microdata set against disclosure quickly.

The Development of **μ**ARGUS

As is explained above, a microdata file should be protected against disclosure in two steps. In the first step some variables are globally recoded. In the second step some values of variables should be locally suppressed. μ -ARGUS, currently under development, will be able to perform these tasks (see Figure 1). μ -ARGUS is a Windows 95 program developed with Borland C++.

Figure 1. -- **μ**ARGUS Functional Design



Metadata

To perform its task, μ -ARGUS should be provided with some extra meta information. At this moment, μ -ARGUS expects the data in a flat ASCII file, so the meta information should contain the regular meta data like the name, the position and the field width of the variables in the data file. Besides this the user needs to specify some additional (SDC-specific) metadata:

- the set of tables to be checked;
- the priority level for local suppression;
- an indication whether a variable has a hierarchical coding scheme -- this knowledge can be used for global recodings, as the truncation of the last digit is a sensible recoding operation for these coding schemes;
- a coding scheme for each variable; and
- a set of alternative codes (recoding schemes) for each key-variable.

The user is not required to specify the coding schemes for all the identifying variables. If the coding

scheme is not specified, μ -ARGUS will inspect the data file and establish the coding scheme itself from the occurring codes.

The level of identification is used to determine the combinations of the variables to be inspected. However, the user is free to determine the set of combinations to be checked, himself.

Generation of Tables and Marginals and Identification of the MINUCs

In order to identify the unsafe combinations and the MINUC's, the first step will be to generate the required tables and marginals. When the data files are available on the PC, the tables will be generated directly on the PC. However, in the case of very large files stored at an other computer (e.g., a UNIX-box), the part of μ -ARGUS that generates the tables can also be transferred to the UNIX-computer to generate the tables there. The ability to run μ -ARGUS on different platforms was the major reason for choosing C++ as our programming language.

When the tables have been generated, it is possible to identify the unsafe combinations. We are now ready to start the process of modifications to yield a safe file.

Global Recoding and Local Suppression

If the number of unsafe combinations is fairly large, the user is advised to first globally recode some variables interactively. A large number of unsafe combinations is an indication that some variables in the microdata set are too detailed in view of the future release of the data set. For instance, region is at the municipality level, whereas it is intended to release the data at a higher hierarchical level, say at the county or province level. To help the user decide which variables to recode and which codes to take into account, μ -ARGUS provides the user with the necessary auxiliary information. After these initial, interactive recodings, the user may decide to let μ -ARGUS eliminate the remaining unsafe combinations automatically. This automated option involves the solution of a complex optimization problem. This problem is being studied by Hurkens and Tiourine of the Technical University of Eindhoven, The Netherlands. Details can be found in Tiourine (1996). In that case, only those MINUCs can be eliminated automatically for which the SDC-metadata file contains alternative codes. The user should specify a stopping criterion, defining which fraction of the set of original MINUCs is allowed to remain, i.e., to be eliminated later by local suppression. The user can continue to experiment with recodings -- both interactive and automatic ones (by undoing them and searching for alternatives) -- until deciding which set of recodings to keep. Recodings that have been interactively established imply that the corresponding metadata (code descriptions, etc.) should be updated as well. If no MINUCs remain the job is finished and the global recodings can be performed on the microdata. However, in general, there are still MINUCs left which have to be eliminated by local suppression.

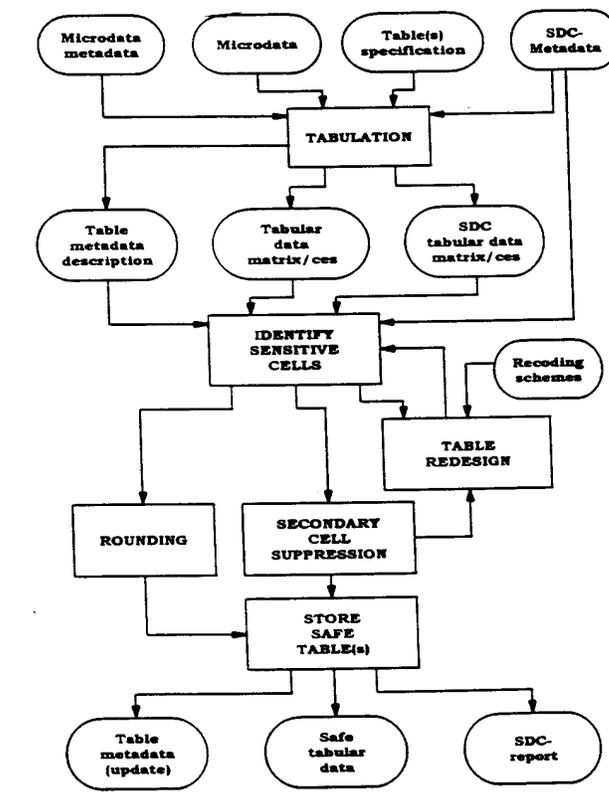
Final Steps

When the above-mentioned steps have been executed, the result is a safe microdata set. The only thing left is to write the safe file to disk and to generate a report and a modified metadata description. In the event that the original data file reside on another computer, μ -ARGUS will generate the necessary recoding information that will be used by a module of μ -ARGUS that runs on that other machine.

The Development of τ -ARGUS

Besides the development of μ -ARGUS for microdata sets, the SDC-Project also plans development of τ -ARGUS. τ -ARGUS is aimed at the disclosure control of tabular data. (See Figure 2.) The theory of tabular disclosure control focuses on the “dominance rule.” This rule states that a cell of a table is unsafe for publication if a few (n) major contributors to a cell are responsible for a certain percentage (p) of the total of that cell. The idea is that, in that case at least, the major contributors themselves can determine with great precision the contributions of the other contributors to that cell. Common choices for n and p are 3 or 70%, but τ -ARGUS will allow the users to specify their own choices. However, some modifications to this dominance rule exist.

Figure 2. -- τ -ARGUS Functional Design



With this “dominance rule” as a starting point, it is easy to identify the sensitive cells, provided that the tabulation package cannot only calculate the cell totals, but also calculates the number of contributors and the individual contributions of the major contributors. Tabulation packages like ABACUS (made by Statistics Netherlands) and the Australian package SuperCross have that capability.

The problem, however, arises when the marginals of the table are published also. It is no longer enough to just suppress the sensitive cells, as they can be easily recalculated using the marginals. Even if it is not possible to exactly recalculate the suppressed cell, it is possible to calculate an interval which contains the suppressed cell. If the size of such an interval is rather small, then the suppressed cell can be estimated rather precisely. This is not acceptable either. Therefore, it is necessary to suppress additional information to ensure that the intervals are sufficiently large. Several solutions are available to protect the information of the sensitive cells:

- combining categories of the spanning variables (table redesign) -- more contributors to a cell tend to protect the information about the individual contributors;
- rounding the table, while preserving the additivity of the marginals; and
- suppressing additional (secondary) cells, to prevent the recalculation of the sensitive (primary) cells to a given approximation.

The calculation of the optimal set (with respect to the loss of information) of secondary cells is a complex OR-problem that is being studied by Fischetti. Details can be found in Fischetti (1996). τ -ARGUS will be built around this solution and take care of the whole process. For instance, in a typical τ -ARGUS session, the user will be presented with the table indicating the primary unsafe cells. The user can then choose the first step. He may decide to combine categories, like the global recoding of μ -ARGUS. The result will be an update of the table with presumably fewer unsafe cells. Eventually, the user will request that the system solve the remaining unsafe cells, by either rounding the table or finding secondary cells to protect the primary cells. The selection of the secondary cells is done so that the recalculation of the primary cells can only yield an interval. The size of these intervals must be larger than a specified minimum. When this has been done, the table will be stored and can be published.

The first version of τ -ARGUS will aim at the disclosure control of one individual table. A more complex situation arises when several tables must be protected consistently, generated from the same data set (linked tables). Then, there will be links between the tables that can be used by intruders to recalculate the sensitive cells. This is a topic of intensive research at this moment. The results from this research will be used to enhance future versions of τ -ARGUS, to take into account links between tables.

References

- De Waal, A.G. and Willenborg, L.C.R.J. (1995a). A View on Statistical Disclosure Control for Microdata, *Survey Methodology*, 22, 1, 95-101, Voorburg: Statistics Netherlands.
- De Waal, A.G. and Willenborg, L.C.R.J. (1995b). Global Recodings and Local Suppressions in Microdata Sets, Report, Voorburg: Statistics Netherlands.

Fischetti, M. and Salazar, J.J. (1996). Models and Algorithms for the Cell Suppression Problem, paper presented at the 3rd International Seminar on Statistical Confidentiality, Bled.

Tiourine, S. (1996). Set Covering Models for Statistical Disclosure Control in Microdata, paper presented at the 3rd International Seminar on Statistical Confidentiality, Bled.

Willenborg, L.C.R.J. (1996). Outline of the SDC Project, paper presented at the 3rd International Seminar on Statistical Confidentiality, Bled.

Willenborg, L.C.R.J. and de Waal, A.G. (1996). *Statistical Disclosure Control in Practice*, New York: Springer-Verlag.