CASC PROJECT

Computational Aspects of Statistical Confidentiality

25 July 2001

Report on the testing of μ-Argus v. 3.1

Giovanni Merola

ISTAT, MPS/D
Via C. Balbo, 16
00184, Roma
Italy

Deliverable No: 6-D1

Report on the testing of μ-Argus v. 3.1

Introduction

μ-Argus is a software that applies state of the art SDC techniques to microdata, being developed within the European Project CASC (Computational Aspects of Statistical Confidentiality) with the final goal of creating a tool usable by NSIs as well as other agencies. This testing of μ-Argus is Work Package 6 of the CASC project and was carried out on the intermediate release 3.1

Testing is a crucial phase in the development of software; this testing has been designed for evaluating μ-Argus especially with respect to its integration in the data production process of statistical institution. That is, having in mind the final user and his/her needs. Hence, the testing has been designed to check for bugs and limitations, collect suggestions for improvements, assess the clarity of the documentation, verify the portability (input/output formats, platform required etc.) and user friendliness of μ-Argus. In this way, scores and comments expressed by testers become a vital source of information to help developers not only to fix bugs and malfunctioning, but also to improve the program towards meeting the requirements of end users. For this reasons testers were selected among potential users of the software, privileging the quality of testing over the quantity of tests run.

Description of the software

μ-Argus 3.1 is a program for applying Statistical Disclosure Control (SDC) to microdata, that is to individual records containing confidential information, that runs under different Microsoft Windows platforms (95/98 and NT/2000).

The program is designed in such a way that SDC can be applied in steps, comparing the effect of different methods and different parameters values in the same run. The first step consists of reading in the data and the metadata. These must be stored in files in a fixed format (.asc and .rda, respectively). The metadata can be edited from within the program using the *modify metadata* option. The combinations of variables to be considered are defined in the *specify combination* window. Also from this window, data can be set in tables. In the second step different criteria of disclosure can be chosen: sampling threshold, individual risk and values for PRAM. The third step is the actual protection of the microdata. In the present version of μ-Argus three different methods are available: global recoding, local suppression and perturbation. The last step is the production of the output which consists of new, protected, data and metadata files and a report in HTML format.

The software should serve as a standard tool for applying different SDC methods to microdata usable by different NSIs.

Outline of the testing set up and questionnaire

Version 3.1 of μ-Argus was tested by 6 testers, one of which on volunteer basis. Also other researchers, external to the CASC project and also from non-European countries, showed interest in testing the program and were included in the list of testers but did not eventually return the questionnaire.

Testers are supposed to have some knowledge of SDC for microdata and access to the existing literature. A set of real data was provided in order to have realistic tests and standardized impressions. These data consisted of 30,000 observations on 9 variables regarding the 11,211

households included in the 1997 survey on household consumptions in Italy. Testers were also encouraged to run the test on other real data produced by their institution.

The questionnaire was designed as a MS Excel worksheet to be sent electronically. A preliminary version was sent to all testers for comments and, a week later, the reviewed version was mailed out, together with instructions and installation files. The completed questionnaire with the results of the testing had to be returned a month later.

The questionnaire is structured in 7 sections, following an ideal microfile protection operation path using the program. The sections were: 1) tester's identity; 2) preliminary issues (installation and data importing); 3) data specification; 4) protection methods; 5) output of protected files; 6) documentation and 7) general remarks. Some questions required an evaluation with four possible scores (I=insufficient, S=sufficient, G=good and E=excellent), other a yes/no answer and the rest just a comment. All questions admitted the possibility of adding comments and suggestions (for scores insufficient or sufficient this was actually required). Further details can be found in the questionnaire with the instructions in Appendix A.

Testing report

An overview of the testing results for each section of the questionnaire will be given in the following section. Detailed summaries of scores and comments for each question can be found in the summary sheet in Appendix B. In this document columns F to K give the frequencies of the scores given to the questions, scores less frequent than 3 are highlighted in Yellow, frequencies of 3 or 4 are highlighted in Orange and higher frequencies are highlighted in Red. Column L gives a mean value obtained either assigning zero to "No" and one to "Yes" or assigning the numerical values 0, 1/3, 2/3 and 1 to the scores I, S, G and E, respectively. In this way, mean values range from 0 to 1, and the closer they are to 1, the closer the mean judgement is to Excellent or Yes and vice versa. Columns M to R indicate whether a tester gave a comment for that question or not (each column refers to a country). Column S contains summaries of the comments given for each question and, finally, column T indicates the countries of which tester did not give an answer for that question.

Overview of the testing results

1) Tester's identity

This section was designed to get an idea of the tester's skills, the equipment they were using for testing and the needs of their institution. Unfortunately, most testers did not give sufficient details on their computer.

Most testers were statistical researchers with good knowledge of SDC for microdata. All tests were run on Pentium III PC's, most of them running Windows NT/2000 and a few Windows 95. There has been one request for a UNIX version of the program.

All the NSI's participating to the testing do release microfiles (mostly for social data), protecting them with global recoding, local suppression and top-bottom coding. Some use also sub-sampling, swapping and perturbation.

2) Preliminary issues (installation and data importing)

This section was designed to gather impressions on issues arising before working with the data. That is, installing the program and data importing. We feel that data and metadata importing is an important issue for users and that the current facility is a bit weak, therefore, a few question insisted on this topic.

Some testers considered a problem the fact that the installation under Windows 2000 requires administrator's privileges; once installed the program could be launched without problems.
The first impression on the graphical appearance is "Good".
Importing the data files gave no problems, however all testers have the need for importing data in formats provided by some commercial programs, especially SAS and MS Excel. The same is true for the metadata file.

The facility for creating the metadata from inside the program was rated sufficiently friendly; nonetheless some testers gave some suggestions for improving it. There has been an almost unanimous agreement on requiring visualizing data summaries in order to check the correctness of the metadata from within the program.

The overall rating for this section was Good on average but better documentation is required.

3) Data specification

Data specification is a crucial step for any statistical analysis; it should be easy and unambiguous in order to set all analysis on firm ground and make comparisons easy. This section deals with editing the metadata, defining the combinations of variables to be considered and specifying the acceptable risk threshold. The whole process leads to computing the tables containing combinations to be inspected by μ-ARGUS. The risk threshold is set by means of an interactive graph created by the *risk specification* option. Since this facility has not been tested before and we felt that it was a bit difficult to use, detailed questions on this topic were asked.

The *specify metadata* window for editing the metadata was found clear and easy to use, however the introduction of more options and features was suggested.
The window for specifying the combinations (*Combination* option) was found complete and useful but a bit difficult to use.
The *automatic specification of tables* option is considered very useful but it is not very clear to all testers.

The *Set Table for Risk Model* option is considered useful. This question was asked to get an idea of how many consider the risk threshold approach useful as, of course, the usefulness of this option depends on whether the approach is used or not.
The output of the "calculate tables", giving the number of unsafe combinations, is not very clear to all testers. It was suggested that it should be corrected in order to take into account the risk threshold approach. The computation time of the tables is slow in some instances.
The summary information provided by the *Show table* option was found very useful and clear (with one exception).
The graphic window for *Risk Specification* option surely needs attention, both as regards the graphical appearance and the content. The graph labels and the control of the sliding bar need improvements. The possibility of specifying manually a numerical value for the threshold was required by many and the idea of a default threshold value is supported by most testers Also the documentation for this topic is in need of improvement.

Overall this section got mostly "Good" marks, however some points need attention.

4) Protection methods

Data protection is the core function of μ–Argus. The applicable methods are: global recoding of variable values, Post Randomization (PRAM) and modification of numeric values. Local suppression is included in next section. Each protection method should be self consisting, therefore we asked specific questions and to comment on the documentation and on the on-line help for each method.

The *global recode* window was rated clear (4 marks of excellence) by most testers. However, it wasn't found as friendly, mainly because the recoding needs a file with the new codes for each variable to be recoded. Some suggested considering only one file for all variables.
The format of the code files (.grc) was considered easy to deal with, however it is not compatible with any commercial software. The error messages for invalid recode list are not clear to all testers and if codes are amended an error crashes the program. The documentation for this method was rated "Good" on average but explanations on how to save code files was required. The on-line help was found worse than the documentation but only minor improvements were suggested.

The *PRAM* method window was neither considered very clear nor very friendly. Also more information and options are required, there also seems to be a wrong indication about the definition of the probability from PRAM. Documentation needs improvement but no suggestions on how to do it were given.

The *modify numeric variable* window got high scores for clarity and friendliness. There are some visual problems, guidance for choosing the new values for top/bottom coding and an undo button were required. The documentation and on-line help seem deficient from the low scores obtained but no suggestion has been made.

The overall score for this section was unanimously *Good* which is telling that the program can already be used for data protection.

5) Output of protected files

The output is the final product of the whole SDC process. It consists of creating a protected data file to be released and a report on the protection of the data. Since the protected file and report could be handed to people not familiar with μ-Argus and SDC, both items should be easy to understand and to explain to others.

The *make suppressed file* window was rated "Good" for clarity, friendliness and information provided by most testers and minor improvements were suggested. Also the manual got mostly "Good" marks. Some testers gave only "sufficient" to the format of the output file with respect to their expectations; unfortunately, though, they did not say why they were not completely satisfied. As for the input file, also for the output files different formats have been strongly requested: mostly Excel and ASCII (comma or tab separated) but also MS Access.

The report produced was found very clear and useful. The information provided was found good but the frequency threshold adopted and number of suppressions over number of records should be also included. The HTML format for the report file satisfies all testers. Saving the file to a different directory seems problematic.

The output section got a unanimous "Good" evaluation with one "excellent".

6) Documentation

Documentation of a program can hardly ever satisfy all users. In the case of scientific programs, such as μ-Argus, a balance must be found between theoretical explanations and guidance to users. Furthermore the scores given may be influenced by the level of theoretical knowledge of the user.

In this section we sought an overall impression of the documentation, where specific critiques can be found in each section.

The on-line help was judged clear and useful on average but not completely exhaustive. However, none of the testers indicated a specific topic that needed improvement. The search facility was judged mostly just "Sufficient" indicating that the index could be expanded to more keywords.

The judgements for the manual were a bit worse than for the on-line help. It was found useful and clear a part from the metadata topic. The manual was found not completely exhaustive by all (large variability in scores though). The topics that deserve more attention are: algorithm used for suppression, risk, treatment of numeric variables and hierarchical variables. The most relevant indications towards improving the manual were: grouping separately the help on SDC theory and that on using the program and lowering the level of the explanations, making them clear to non-advanced users.

Overall the documentation got two "Sufficient" scores and all others "Good", indicating that some improvement is needed.

7) General remarks

This section asks testers to express their overall opinion on the program. Questions regarding the organization of the menu options, intuitiveness of the ordering of the steps for carrying out the protection of a data file and overall functionality are included. Testers are also asked if they would advice their institution to adopt it "were it a commercial program". This question was asked in order to have the tester change perspective.

There is variability in the judgements over the organization and homogeneity of the menu items. Some testers judged them consistently "Excellent", others found them just "Sufficient" but did not tell how they would see them improved. The overall score was "Good" on average with some marks of excellence.

The sequence of steps needed to apply the protection methods was judged from good to excellent with one exception, some kind of guidance on choices and effects of previous steps was required.

The application of SDC techniques was judged to be offered in a clear and informed way. A few testers would prefer a "step to step" (or a wizard) over the drop-down menus. The overall score for the application of SDC techniques was unanimously "Good".

The guidance provided to the user was also found adequate and only one tester gave low marks to the labelling. Among requests for improvement are: addition of "Undo" buttons in some windows and "Help" buttons on every window. The introduction of a tutorial "Getting started with μ-Argus" was also suggested.

The program has had conflict with other running applications for only one tester, who did not, however, specify which program caused the problem. There has not been registered significant slowing on the system while running the program. There seem to be serious problems with the handling of errors in the metadata file and in the file for recoding. Some testers experienced slowness in reading in data and computing the tables for several variables.

The computational speed was rated Sufficient/Good.

The overall functionality of the program was rated Sufficient/Good and seems in need of improvement. There have been registered some errors and warnings during normal working session and some errors cause crashes, when a warning would be sufficient. Also, the program is slow when dealing with several variables.

All testers concluded that the program could be adopted by their institution as a standard tool for SDC and that they would all recommend buying it to their institution. The overall rating was unanimously "*Good*".

Conclusions

The testing of μ -Argus pointed out some minor bugs and inefficiencies. Some testers gave some suggestions on how they would like to see the program improved. It seems that the documentation and the handling of errors should be improved. However the comments and judgements showed that the program is considered a good, reliable tool that could be already adopted by NSIs.