CONFERENCE OF EUROPEAN STATISTICIANS

Expert Meeting on Statistical Data Confidentiality

1-3 December 2021, Poland

Automatic checking of research outputs.

Marco Stocchi (Eurostat)

Abstract

Checking research output for disclosure risks is an activity traditionally performed in statistical institutes and departments by specialist personnel. It is a time-consuming task and it requires significant resource allocation.

In an effort to reduce the workload of the offices which carry out the output checks, Eurostat commissioned a project to investigate the feasibility of a software that does confidentiality checks on a set of algorithms popularly used by the researchers. The project was carried out by GOPA - Eurostat contractor for the framework contract on Methodological Support and by the team of experts of the University of West England: Elizabeth Green, Felix Ritchie and James Smith.

The project resulted in a proof of concept in STATA® - a tool named ACRO (Automatic Checking of Research Output) and the Eurostat Statistical Working Paper [1] describing the software and discussing some general aspects of (automatic) output checking. The present document is a shorter version of the above mentioned working paper. It describes current functionalities of the tool and proposes some further developments.

Given the open source code of the tool, publicly available on a Github repository (https://github.com/eurostat/ACRO) and its permissive licensing terms, we welcome the involvement of a community of practitioners to elaborate on possible feature extensions and to contribute to its development.

Automatic checking of research output (ACRO)

M. Stocchi*, A. Bujnowska*

October 2021

Abstract

Checking research output for disclosure risks is an activity traditionally performed in statistical institutes and departments by specialist personnel. It is a time-consuming task and it requires significant resource allocation.

In an effort to reduce the workload of the offices which carry out the output checks, Eurostat commissioned a project to investigate the feasibility of a software that does confidentiality checks on a set of algorithms popularly used by the researchers. The project was carried out by GOPA - Eurostat contractor for the framework contract on Methodological Support and by the team of experts of the University of West England: Elizabeth Green, Felix Ritchie and James Smith.

The project resulted in a proof of concept in STATA® - a tool named ACRO (Automatic Checking of Research Output) and the Eurostat Statistical Working Paper[1] describing the software and discussing some general aspects of (automatic) output checking. The present document is a shorter version of the above mentioned working paper. It describes current functionalities of the tool and proposes some further developments.

Given the open source code of the tool, publicly available on a Github repository (`https://github.com/eurostat/ACRO`) and its permissive licensing terms, we welcome the involvement of a community of practitioners to elaborate on possible feature extensions and to contribute to its development.

Keywords— confidentiality, output checking, statistical disclosure control

1 Introduction

The access to highly confidential data is provided through research data centres, which allow the researchers to process the data in a secure environment (endowed with both physical and information security features). Moreover, some statistical offices already allow remote access to their confidential data directly from the computers of the researchers[1].

*Eurostat, Unit of Methodology and Innovation in Official Statistics

[1]Eurostat is currently setting up a remote access system to the European secure use files, from accredited end-points located at the premises of research entities.

The output checking problem is similar regardless how the access takes place - on-site or remotely. The results of the work produced in research data centre must be checked to avoid release of any confidential data.

At the best of our knowledge, conceptually the research output checking process - herein after ROC, can be categorized as follows: *i*) ROC entirely made by the output checker personnel (thus without the support of automated systems); *ii*) ROC made by the output checker with some expert system supporting the decision making; *iii*) ROC made in part by human checking activity and in part by machines deployed to the purpose of direct scrutiny/rejection of unsafe output; *iv*) ROC entirely performed by automated processes. Any automation of ROC is usually associated with some upward checks of codes and routines used by researchers and thus is often considered as limitation.

The first category of ROC - manual checks of research results by output checker - is the one usually adopted by the statistical offices and, despite its work intensiveness and lack of scalability, it is believed to be highly secure. The researcher has access to full data-sets and submits the results for checking once the data analysis is completed.

The second category defines a ROC process composed of an initial automatic analysis of the research output (aiming at running the checks based on known theoretical statistical disclosure control - SDC rules [3]). Conclusions will be drawn by the output checker once analyzed the results of the automatic checks. We classify ACRO as a tool belonging to this category.

The third category comprehends systems that receive some sort of initial input from the researchers, such as the code or script that will be run on the micro-data; and successively performing an automatic screening based on the knowledge of unsafe or problematic routines that are banned by default, thus rejecting the possibility of running unsafe computations on the data in the first place. One such example is the remote execution data access system named LISSY[2].

The fourth category, less explored by the literature in the field of ROC, is conceptually featured by autonomous systems endowed with semantic knowledge discovery capabilities and extensive access to different data sources, possibly including privately held data. Such systems would intersect and use information found in different origins, to the purpose of a disclosure risk analysis that extends beyond the current capabilities of the traditional SDC implementations.

This paper is organized as follows: Sec.2 describes the functional features of the ACRO tool; Sec.3 reviews the main technical implementation aspects of the software and its workflow; Sec.4 draws conclusions and indicates further work directions aimed at both the feature extension and the portability enhancement of ACRO.

2 Design

In order to design the tool, the project team used the *MoSCoW* design framework (must-have, should-have, could-have, won't have [4]). Tab.1 presents the ACRO functionalities, in terms of type of statistics and operational features, mapped to each respective *MoSCoW* flag. The choices of elements to be included or excluded were agreed with Eurostat in advance, striving to find a viable compromise between software utility and development costs.

[2]https://www.lisdatacenter.org/

Flag	Statistics	Operations
Must have [a]	Tabulation of descriptives [b]	No loss of information except where suppressed
	Linear and non-linear estimation	Primary suppression and recalculation of totals
	Weights	Exceptions
		User choice which statistics to submit for checking
		User ability to overwrite previous outputs
		Easier and faster checking for output checker
Should have [c]		Automatic suppression at the user's request
		Minimal training for users
		Minimal need for users to adjust their code
		Dataset-specific SDC criteria
Could have [d]	Graphs	Minimal training for output checker
		Minimal setup on user systems
		Output which has added value for the user
		Disclosure by differencing
Will not have [e]	Other	Secondary suppression

Table 1: ACRO functionalities. [a] required functionality; [b] frequency, percentiles, moments, maxima and minima; [c] absence will notably diminish functionality; [d] presence will enhance functionality; [e] not necessary, or too complex, or time consuming implementation task.

Tabulation and common estimators were considered essential functionalities of ACRO. Graphs were attributed a lower priority, since they can be considered as a different presentation of tabular data; thus addressing the disclosure risk analysis in the tabulation phase was believed to suffice in the context of a proof of concept development.

The tool was expected to deal with primary suppression, with totals recomputed as necessary. Secondary suppression was not considered a good SDC solution for research output, being *i*) difficult to achieve, even using dedicated table-suppression software; *ii*) error prone and more likely to increase the risk of disclosure by differencing [1]. Whenever outputs would fail the primary disclosure checks, the user would have to redesign the table in order to remove any primary confidential cell. This design feature circumvents the need for secondary suppression routines.

ACRO would also be endowed with mechanism suitable to request 'exceptions', in line with *principle-based output statistical disclosure control*, to allow researcher to explain and justify why particular results are not disclosive.

Minimal set-up was thought to be important to encourage the adoption of the tool by the users, who would also be incentivized to use it in order to shorten the output clearance times.

Keeping reasonable the amount of required training was retained, in accordance with the general objectives of the tool to reduce the overall amount of resources allocated by the statistical offices to the output checking activities.

3 Implementation

ACRO was developed as a collection of STATA routines. This was largely determined by the skill-set of the authoring team, but also because STATA was felt to be more readable than R (its main competitor in research environments). Python was considered as a more future-proof alternative. However, being the team less familiar with the Python programming language, it was unclear how it could be seamlessly integrated with other R or STATA existing tools.

ACRO requires a set of files (in '`.ado`' format), along with an Excel macro-enabled template, to be uploaded in a folder to which the researcher has read-only access. To initialise ACRO, the researcher runs a set-up instruction. Before each output to be reviewed, he or she prefixes with '`safe`' the command to be run, with some optional parameters; otherwise, the syntax is unchanged. The commands are sent to an Excel file; the researcher receives messages about the SDC check, as well as being able to check the Excel file.

Finally, the researcher calls the command '`finalise`', which instructs ACRO to prepare a Microsoft® Excel® file for the successive review of the output checker, who is then able to see all the requested outputs: *i*) those which have failed; *ii*) the ones that passed; *iii*) those which have passed after the suppression; *iv*) those which would have failed, but an exception was requested.

An Excel macro was implemented in order to allow the output checker to review the exceptions and quickly remove those retained not acceptable. ACRO's functionality was plotted in flow diagrams to understand the necessary decision points.

The code was delivered on all of the *must*, *should*, *could* described above, with the following remarks: *i*) no satisfactory solution was found for disclosure by differencing (the main idea is to avoid any primary confidential cells), *ii*) graphs were included so that all the requested output could be in one file for review; *iii*) percentiles, other

Issue	Addressed	Description
Identification and association	Fully addressed	The organisation has full freedom to set the appropriate SDC limits consistent with its risk preferences. Exceptions can be granted, subject to manual review.
Primary disclosure (small numbers and dominance)	Fully addressed	This is a deterministic operation, given the SDC rules. ACRO is likely to check for primary disclosure better than a human as checks which are difficult for humans (such as dominance) can be easily automated.
Class disclosure	Not addressed	Empty and full cells are allowed in tables: as data are usually (sub)samples, absence of information is usually uninformative. The alternative, removing tables with empty cells, was felt to be too restrictive.
Sample or population?	Not addressed	ACRO only understands counts in cells, not their true representation in the world. Therefore, it over-protects.
Secondary disclosure	Not addressed	As noted above, there is no general solution to this problem.
Automated (semantic) reasoning	Not addressed	ACRO uses no semantic information to improve its decision-making.
'Safe' and 'unsafe' statistics	Fully addressed	This is directly implemented (it greatly simplifies the checking problem), albeit currently for a small range of statistics.
Actual versus potential disclosure	Not addressed	ACRO is a blunt instrument, assuming every potential risk is an actual one; as such it over-protects.

Table 2: Statistical issues addressed by ACRO. While issues such as identification and association, primary disclosure, safe and unsafe statistics and actual versus potential disclosure are fully addressed by ACRO; class disclosure, sample versus population, secondary disclosure and automated reasoning are not addressed.

than medians, were not implemented: these were felt to be a lower priority, and were omitted due to time constraints; *iv*) maxima and minima are banned by default (i.e. requests to report the maximum/minimum are either rejected, or suppressed if automatic suppression is chosen by the user).

Tab.2 presents the statistical issues currently addressed by the proof of concept. While issues such as identification and association, primary disclosure, safe and unsafe statistics [2] and actual versus potential disclosure are fully addressed by ACRO; class disclosure, sample versus population, secondary disclosure and automated reasoning remain unaddressed.

The technical reader can also find illustrative flow charts underlying the mid-program functionality in the Appendix 1 to the Eurostat Working paper [1].

4 Conclusions and further work directions

After implementation, both the code and user/manager guides were distributed to several Statistical Offices in order to perform preliminary tests. The most common feedback received can be summarised as follows:

- ACRO exhibits a good potential to reduce manual output checking;
- it is not as user intuitive as the team planned, despite the documentation contained in the guides; however, it is easy to familiarize with ACRO in a relatively short amount of time;
- implementing a wider range of functions (such as percentiles and simple checks on graphs) would substantially improve its utility;
- porting the software to different programming languages and/or platforms (e.g. R, Python, SAS®) is necessary in order to successfully deploy ACRO throughout the Research Data Centers.

At present the ACRO pilot is released as version alpha, a working proof of concept, and uploaded on a Git repository[3] to the benefit of the general public. Moving to the 'beta' version (an approximation to a production tool, albeit still running in STATA), requires *1)* expansion of the programmed features to cover all of the most common commands, where possible; *2)* development of a user engagement program to support uptake; *3)* an understanding of how the tool can integrate with organisational processes.

As one of the design goals was to make something that researchers would use by choice, a 'beta' version will require extensive acceptance testing. The acceptance of the output checkers is also important. One future development is to adapt the current pseudo-code in order to formulate other versions of the tool which could be run across a number of different platforms (e.g. SAS® or MatLab®); alternatively, porting the tool to other popular programming languages (e.g. R, Python) would also benefit the community of official statistics.

References

[1] Ritchie F., Green E., Smith J. (2021): "Automatic Checking of Research Outputs (ACRO): a tool for dynamic disclosure checks". Eurostat Statistical Working

[3] https://github.com/eurostat/ACRO

Papers series. Luxembourg: Publications Office of the European Union, 2021. doi: 10.2785/618842.

[2] Bond S., Brandt M., de Wolf P-P. (2015): "Guidelines for Output Checking", Eurostat. (`https://ec.europa.eu/eurostat/cros/content/guidelines-output-checking_en`).

[3] Hundepool A., Domingo-Ferrer J., Franconi L., Giessing S., Schulte Nordholt E., Spicer K. and de Wolf P-P. (2012): "Statistical Disclosure Control". Wiley.

[4] Clegg, D, Barker, R. (1994): "Case method fast-track: a RAD approach". Addison-Wesley Longman Publishing Co.,Inc.