Ex Post Survey Harmonization with retroharmonize

Early Draft. DOI: 10.5281/zenodo.6536273

Daniel Antal, University of Amsterdam
Annamária Tátrai, Eötvös Lóránd University Leo Lahti, University of Turku
Pyry Kantanen, University of Turku

2022-05-10 16:48:02

Abstract

A short (ca. 100 word) summary of the software being described: what problem the software addresses, how it was implemented and architected, where it is stored, and its reuse potential.

Survey data harmonization refers to procedures that combine survey data from different sources, that is, they harmonize survey data. Survey users often improve the data comparability or the inferential capacity of multiple surveys conducted in different periods of time, or in different geographical locations, potentially using different languages. This approach is known as ex-post output harmonization, or, simply, ex-post or retrospective harmonization. Ex ante, prospective or input harmonization on the other hand refers to practices of data producers to give more opportunities for retrospective harmonization.

The retroharmonize package support various data processing, documentation, file/type conversion aspects of various survey harmonization workflows. Our examples are made with data from ex ante harmonized surveys—as our examples shows, retrospective harmonization remains a challenging task even when the data producer was designing the surveys with the purpose of data harmonization.

Keywords survey data; survey harmonization; statistical matching, open data Keywords should make it easy to identify who and what the software will be useful for.

Introduction

An overview of the software, how it was produced, and the research for which it has been used, including references to relevant research articles. A short comparison with software which implements similar functionality should be included in this section.

Surveys, i.e., systematic primary observation and data collections are important data sources of both social and natural sciences. They are in most cases the primary data sources of scientific research. Drawing information from several surveys, conducted in different locations or in different time can greatly enhance the inferential capacity of the surveys, but it requires significant data processing and statistical processing work.

"To ensure that answers from respondents surveyed in different settings carry minimal methodological errors and biases and can be meaningfully compared, both data producers and secondary users combine surveys from different sources, that is, they harmonize survey data. Generally, they do so at different stages of the survey lifecycle. Data producers mostly employ harmonization ex-ante, when designing and implementing comparative studies (input harmonization) and when processing the survey data in preparation for their public release (ex-ante output harmonization). [...] Secondary users apply harmonization methods retrospectively to already released data files." (Wysmułek, Tomescu-Dubrow, and Kwak 2022)

Survey data harmonization refers to procedures that improve the data comparability or the inferential capacity of multiple surveys conducted in different periods of time, or in different geographical locations, using different languages. Retrospective harmonization, or "survey recycling" is a practice for integrating information from two or more data sources, and to create a from several survey documentations, several codebooks, and several data tables, single, consolidated documents, and tables. (Slomczynski and Tomescu-Dubrow 2018)

Survey harmonization is closely related to the concept of statistical matching, also known as data fusion or data matching, the practice of "drawing information piecewise from different independent sample surveys," particularly the bottom-up approaches to these problems. (D'Orazio, Di Zio, and Scanu 2006, p1) Statistical matching takes survey recycling a step further, aiming to improve the statistical inference capacities of the joined dataset, for example, with creating a unified weighting for the variables. In our software package we drew the line where the joined datasets, joined codebooks, and a general description is delivered: that is where our retroharmonize package aims to help.

In the R statistical ecosystem there is a mature package for statistical matching, (D'Orazio 2020), which gives a programmatic solution to the long contributions from the author in data matching. However, we have not found a similarly comprehensive solution to retrospective survey harmonization. There are several packages which fulfil partial tasks required to achieve a joined dataset and joined codebook, but not in a unified workflow and with all the ingredients present from importing an existing survey, processing both its data and metadata, and placing it into a more suitable format, augmented by variables necessary join data from several sources (such as an observation identifier that is truly unique among two or more original datasets, or potentially a consistent, new, post-stratification weight.)

Our software builds on many earlier released open source software solutions that facilitate working with surveys. These tools were built for single surveys, therefore they do not treat the problems of conflicting coding of the same variable, conflicting naming of the same concept, or coercion problems when the same information is stored in different ways, gets imported into different R classes, and joining may result in unexpected output.

The DDI Alliance has released the Structured Data Transformation Language (SDTL) has been designed for standardising an intermediate language for representing data transformation commands. Statistical analysis packages (e.g., SPSS, Stata, SAS, and R) provide similar functionality, but each one has its own proprietary language (Alter et al. 2020). Since the first, peer reviewed CRAN release of retroharmonize, the DDIwR package has been developed, but not yet released on CRAN and not yet fully documented (Dusa 2021). DDIwR solves similar problems that we solved with the introduction of the inherited s3 class labelled_spss_survey with a far more general and ambitious goal. We foresee that in the future we will create full interoperability with that package, and indirectly with the survey harmonization efforts of the international DDI Alliance.

Implementation and architecture

> How the software was implemented, with details of the architecture where relevant. Use of relevant diagrams is appropriate. Please also describe any variants and associated implementation differences.

Retroharmonize was developed over several years with implementing more and more harmonization tasks in a reproducible manner (*The Practice of Reproducible Research* 2018), working with actual data that was *ex ante* harmonized on various levels, first with the European Eurobarometer series, then adding Afrobarometer, Lationbarometro and private surveys. These international survey research programs provide access to their harmonized surveys in "waves." Usually, they call a way a set of ex ante harmonized surveys (containing the same questionnaire in several languages) in one data collection period. Our added value has been that we further harmonize data among waves (when data is not fully ex ante harmonized and requires ex post or retrospective harmonization.) While ex ante harmonized surveys are designed with the aim of ex post harmonization, in our experience, surveys taken across time (the Eurobarometer survey have an almost

50-years of history), using different software solutions available at the time, creates plenty of retrospective harmonization tasks even in these cases.

Our work is building on, an extending many elements of the tidyverse R software packages that "share an underlying design philosophy, grammar, and data structures" of tidy data (Wickham et al. 2019). "Tidy datasets are easy to manipulate, model and visualize, and have a specific structure: each variable is a column, each observation is a row, and each type of observational unit is a table." (Wickham 2014) Processing survey data, usually stored in more complex sturctures of data and descriptive metadata (coding) into tidy forms is essential for harmonizing variable names, labels, identifiers, and add new auxiliary data, such as unified weights, for further statistical analysis. Tidy datasets lend themselves to easier unit testing, too.

1. **Importing survey data**: Most surveys are stored in some special, statistical file format, usually SPSS's .sav or STATA's dta file. Our package depends on, and adds functionality to two, widely used R packages that can manage the data and metadata of a single survey, but do not check metadata inconsisentencies when joining data from several surveys. Our importing functions add functionality to the `read_sav` and `read_dta` functions of the haven package (Wickham and Miller 2021). The haven package in itself builds upon the `labelled` S3 class and its methods (Larmarange 2021). We had to create an inherited class from `labelled` and `labelled_spss` to add the necessary consistency checks to metadata and coercion rules, when the data arrives from different datasets and files. For example, in Survey 1 the data may be read as a factor variable into R, and with the same coding, read as a numeric into R from Survey 2. Naive concatenation can have unintended effects, or stop with an error message in case of a mismatch between `labelled` and `labelled_spss`.

2. **Concept harmonization**: We created functions that help understanding what concepts are represented in the survey, and how they can be harmonized. Our `metadata_create` function creates an exploratory mapping of the selected files to overview the basic statistical properties of the data, the idiosyncracies of the metadata, and potential R type conflicts in coercion.

3. **Crosswalk**: we use a simple crosswalk schema, or crossswalk table to define variable harmonization, variable code and label harmonization, and R type conversion steps in the output harmonization. There are several R packages that are tacking a similar problem. The R package crosswalkr strongly overlaps with our approach to crosswalk, but critically, it does not work with SPSS files (Skinner 2020), which is the most commonly used data file type storing survey data in openly availalbe international harmonized survey programs.

4. **Type conversion**: When the harmonized output is not a simple harmonized dataset, but a harmonized statistic or a harmonized indicator, the harmonized dataset needs further statistical processing. To unleash the vast potential of R's statistical package ecosystems, it is important to harmonize the data into R's base classes, numeric for numerical statistics and factor for categorical statististc. (Often the same data can be given both numeric and factor representation.) Our inherited tidyverse s3 classes, survey and labelled_spss_survey were designed to make these procedures well documented, reproducible, and unambiguous. The most frequently used file format for survey data is SPSS, which uses a code/label representation that, when imported without our added consistency checks, can be coerced to numerical or factor formats that result in serious logical errors. (This is mainly due to the cause that responses which must be translated to NA_real_ in numeric format are always coded as integers in SPSS–and their interger values and labels indicating missigness are not harmonized.)

5. **Codebook creation**: Our aim is to create high value output that can be further developed with statistical matching, and eventually analysed in a scientific or well-docuemnted policy/business workflow. This means that we re-code and retain as much descriptive and processing metadata as possible, and eventually allow the exporting of a consistent codebook.

Our package has a rich, long-form, vignette article documentation with examples on how to perform survey harmonization Eurostat, Afrobarometer, and Arab Barometer survey files.

Quality control

Detail the level of testing that has been carried out on the code (e.g. unit, functional, load etc.), and in which environments. If not already included in the software documentation, provide details of how a user could quickly understand if the software is working (e.g. providing examples of running the software with sample input and output data). Retroharmonize was extensively tested on privately conducted surveys, and three large, international, ex ante harmonized survey programs (questionnaire-based social science research aimed for ex post or retrospective harmonization across countries and years): Eurobarometer, Afrobarometer and Latinobarometro.

Our aim was to create a package that can accompany a social scientist working with surveys on a personal computer. We soon realized that working with ex ante harmonized surveys may potentially lead plenty of resources, particularly because the typical file format used for surveys, SPSS, due to its dual data-metadata coding, is not very efficiently imported with tidyverse's haven to R. All important functions were designed to work either with a list of surveys being documented, subsetted, renamed, recoded, or sequentially. These functions can take an optional survey_paths (full path) or survey_path and import_path (directory) input, in which case each task is performed sequentially. The optinal export_path, when given, saves the sequentially intermediate or final outputs with saveRDS as R objects.

Our aim was to create a package that can accompany a social scientist working with surveys on a personal computer. We soon realized that working with ex ante harmonized surveys may potentially lead plenty of resources, particularly because the typical file format used for surveys, SPSS, due to its dual data-metadata coding, is not very efficiently imported with tidyverse's haven to R. All important functions were designed to work either with a list of surveys being documented, subsetted, renamed, recoded, or sequentially. These functions can take an optional survey_paths (full path) or survey_path and import_path (directory) input, in which case each task is performed sequentially. The optinal export_path, when given, saves the sequentially intermediate or final outputs with saveRDS as R objects.

This allows a much faster looping when sufficient memory is present, or a slower looping over files. We also included simple functions for resource planning, and tutorials to show the optimal workflow (usually subsetting of many SPSS files should be done sequentially but the later stages of harmonization can take place in memory.)

For unit testing, we included in the R package three subsets of published Eurobarometer surveys. The package's unit testing consists of about 130-unit tests made with this real-life survey excepts.

Availability

Operating system

The retroharmonize R package is tested to run on several different operating systems. According to Microsoft, R 3.5.0 is tested and guaranteed to run on the following platforms: Windows® 7.0 SP1 or later, Ubuntu 14.04 or later, CentOS / Red Hat Enterprise Linux 6.5 or later, SUSE Linux Enterprise Server 11 or later, Mac OS X El Capitan (10.11) or later macOS versions.

Programming language

The retroharmonize R package depends on R version 3.5.0 or higher.

Additional system requirements

According to Microsoft, minimum system requirements for R 3.5.0 are 64-bit processor with x86-compatible architecture, 250 MB of free disk space and at least 1 GB of RAM. These requirements are met by most computers sold in the last 10 years.

On more modern R versions, the package is tested to run on a wide variety of operating systems and system configurations, including ARM-based Macs.

Dependencies

The retroharmonize R package depends only on R (version 3.5.0 or greater). The package imports functions from the following packages: * R Core packages: methods, stats, utils; * tidyverse packages: dplyr (1.0.0 or greater), glue, haven, magrittr, stringr, tibble, tidyr, purrr; * R infrastructure (r-lib) packages: fs, here, pillar, rlang, tidyselect, vctrs; and * other R packages: assertthat, labelled, snakecase

The retroharmonize R package is practically a very thorough extension of the R tidyverse packages: it depends on haven (and labelled) for working with coded survey files. It uses dplyr, tidyr (and their common, deep level rlang, vctrs) dependencies for variable manipulation within a single survey (preparation for harmonization) and purrr for functional programming task with several surveys.

List of contributors

Please list anyone who helped to create the software (who may also not be an author of this paper), including their roles and affiliations.

Marta Kolcynska as a survey harmonization expert contributed to the conceptual development of the first documentation, building the first use cases and exploring various survey harmonization workflows that may need a reproducible and computational support.

Software location:

Archive (e.g. institutional repository, general repository) (required – please see instructions on journal website for depositing archive copy of software in a suitable repository) Name: CRAN Persistent identifier: https://CRAN.R-project.org/package=retroharmonize Licence: GPL-3 Publisher: Daniel Antal Version published: 0.2.0 Date published: 02/11/21 Code repository (e.g. SourceForge, GitHub etc.) (required) Name: retroharmonize Identifier: https://github.com/rOpenGov/retroharmonize Licence: GPL-3 Date published: 15/12/21 Emulation environment (if appropriate) Name: The name of the emulation environment Identifier: The identifier (or URI) used by the emulator Licence: Open license under which the software is licensed here Date published: dd/mm/yy

Language

Language of repository, software and supporting files

English

Discussion

Reuse potential

> Please describe in as much detail as possible the ways in which the software could be reused by other researchers both within and outside of your field. This should include the use cases for the software, and also details of how the software might be modified or extended (including how contributors should contact you) if appropriate. Also you must include details of what support mechanisms are in place for this software (even if there is no support).

The retroharmonize R package aims to provide a versatile support for various survey harmonization workflows. Because surveys are so fundamental to quantitative social science research and play an important role in many natural science fields, not to mention commercial applications of market research or pharmaceutical research, the package's main reuse potential is to be a foundation of further reproducible research software aimed to automate research and harmonization aspects of specific survey programs.

The authors of this package started the development work to be able to harmonize surveys from harmonized data collections of the European Union: namely the Eurobarometer and AES surveys programs. After working with various surveys (also outside these programs) it became clear that retroharmonize should aim to be a common demoninator to a family of similar software that solves more specific problems. The world's largest and oldest international public policy survey series, Eurobarometer. This program alone has conducted already thousands of surveys in more than 20 natural languages over more than 40 years, following various documentation, data management, coding practices that were not independent of the software tools available over this long period of time. The first version of retroharmonize was separated to the retroharmonize and the eurobarometer R packages – retroharmonize providing a more general framework that has been able to serve Eurobarometer's, Afrobarometer's and the Arab Barometer's different needs. Furthermore, the package could be linked to the broader R ecosystem that provides interoperable packages for retrieving data from Eurostat or other statistical authors (Lahti et al. 2017) for downstream harmonization and analysis.

Something about the limitations

Despite the mature documentation and tested functionality, each data source will require customized treatment. The methods of the *retroharmonize* package will provide the basis for building reproducible workflows, but the fluent use will require good knowledge of the package capabilities.

In our experience, each survey program has an established vocabulary, standardized language, and similarity in coding errors. Creating codebooks, crosswalk schemas can be greatly helped by custom dictionaries, which can be published as data packages, for example, for the Eurobarometer or the Afrobarometer survey program.

Future work could include improvements on creating connecting data packages that contain dictionaries and even common coding errors for large survey programs. Such efforts could overlap with prospective harmonization work, such as the creation of standardized questionnaire items, for example, the long running Cross-National Equivalent File (CNEF) project that aggregates both direct and constructed variables from various country's panel surveys into one easy to access location (re3data.org 2017).

Survey harmonization is often addressed in cross-country comparison, however, we believe that comparisons among sub-national statistics gained from surveys offers even more insight. For example, individual national surveys within the Eurobarometer program about the United Kingdom contain a separate sample for Norhern Ireland (which is in the European territorial nomenclature a region) and contain subsamples of Great Britain's regions, for example, Wales. Apart from Northern Ireland, the UK sub-national samples are not designed to be representative, nontheless, we can create large longitudional files from respondents only from Wales. Connecting harmonized data output with small area statistics can substantially increase the ecological inferential capabiltiy from survey data – in the European Union only, instead of working with 27 member state's data, we could create hundreds of subsamples.

The challenges in data harmonization and integration have been widely recognized in computational social sciences and digital humanities (Mäkelä et al. 2020). Thus, in our view, the retroharmonize package has the potential to become a general and widely used supporting software for more specific codes aimed at harmonizing surveys based first on questionnaires, later on different data inputs, such as price scanning, laboratory tests, and other standardized, discrete inputs that are carried out in different locations, with different recording tools, and with different coding (for example, because of natural languages differences, as it is the case in the social science surveys used for the testing of our software.)

Acknowledgements

Please add any relevant acknowledgements to anyone else who supported the project in which the software was created, but did not work directly on the software itself.

LL and PK were supported by Academy of Finland (decision 295741).

Funding statement

There was no funding available for the development of this software.

Competing interests

"The authors declare that they have no competing interests."

References

Alter, George, Darrell Donakowski, Jack Gager, Pascal Heus, Carson Hunter, Sanda Ionescu, Jeremy Iverson, et al. 2020. "Provenance Metadata for Statistical Data: An Introduction to Structured Data Transformation Language (SDTL)." *IASSIST Quarterly* 44 (4). https://doi.org/10.29173/iq983.

D'Orazio, Marcello. 2020. *StatMatch: Statistical Matching or Data Fusion*. https://CRAN.R-project.org/package=StatMatch.

D'Orazio, Marcello, Marco Di Zio, and Mauro Scanu. 2006. *Statistical Matching: Theory and Practice*.

Dusa, Adrian. 2021. "DDIwR: DDI with r." https://CRAN.R-project.org/package=DDIwR.

Lahti, Leo, Janne Huovari, Markus Kainu, and Przemysław Biecek. 2017. "Retrieval and Analysis of Eurostat Open Data with the Eurostat Package." *The R Journal* 9 (1): 385–92. https://github.com/openresearchlabs/openresearchlabs.github.io/blob/master/content/publication_resources/papers/2017-Lahti-RJournal.pdf.

Larmarange, Joseph. 2021. "Labelled: Manipulating Labelled Data." https://CRAN.R-project.org/package=labelled.

Mäkelä, Eetu, Krista Lagus, Leo Lahti, Tanja Säily, Mikko Tolonen, Mika Hämäläinen, Samuli Kaislaniemi, and Terttu Nevalainen. 2020. "Wrangling with Non-Standard Data." In *Proc. Digital Humanities in the Nordic Countries*, edited by Sanita Reinsone, Inguna Skadiņa, Anda Baklāne, and Jānis Daugavietis, 2612:81–96. CEUR Workshop Proceedings. https://github.com/openresearchlabs/openresearchlabs.github.io/blob/master/content/publication_resources/papers/2020-Makela-DHN.pdf.

re3data.org. 2017. "Cross National Equivalent File CNEF Editing Status 2017-11-21." re3data.org - Registry of Research Data Repositories. 2017. https://doi.org/http://doi.org/10.17616/R3MM2B.

Skinner, Benjamin. 2020. "Crosswalkr: Rename and Encode Data Frames Using External Crosswalk Files." https://CRAN.R-project.org/package=crosswalkr.

Slomczynski, Kazimierz M., and Irina Tomescu-Dubrow. 2018. "Basic Principles of Survey Data Recycling." In *Advances in Comparative Survey Methods*, 937–62. John Wiley & Sons, Ltd. https://doi.org/https://doi.org/10.1002/9781118884997.ch43.

The Practice of Reproducible Research. 2018. 1st ed. University of California Press. http://www.jstor.org/stable/10.1525/j.ctv1wxsc7.

Wickham, Hadley. 2014. "Tidy Data." *Journal of Statistical Software* 59 (10): 1–23. https://doi.org/10.18637/jss.v059.i10.

Wickham, Hadley, Mara Averick, Jennifer Bryan, Winston Chang, Lucy D'Agostino McGowan, Romain François, Garrett Grolemund, et al. 2019. "Welcome to the tidyverse." *Journal of Open Source Software* 4 (43): 1686. https://doi.org/10.21105/joss.01686.

Wickham, Hadley, and Evan Miller. 2021. "Haven: Import and Export 'SPSS', 'Stata' and 'SAS' Files." https://CRAN.R-project.org/package=haven.

Wysmułek, Ilona, Irina Tomescu-Dubrow, and Joonghyun Kwak. 2022. "Ex-Post Harmonization of Cross-National Survey Data: Advances in Methodological and Substantive Inquiries." *Quality & Quantity* 56 (3): 1701–8. https://doi.org/10.1007/s11135-021-01187-7.