Maciej Beręsewicz, PhD
Poznań University of Economics and Business
Department of Statistics
Poland
maciej.beresewicz@ue.poznan.pl

Cover Letter April 23, 2025

Dear Editors of the Journal of Statistical Software,

we are writing to resubmit an revised manuscript (no. 5753) of our article entitled ***nonprobsvy*** – *An R package for modern methods for non-probability surveys* for review to the Journal of Statistical Software. Replies to the comments are provided at the end of this cover letter.

Official statistics traditionally rely on probability surveys, censuses, or administrative registers that cover the entire population. However, rising non-response rates and survey costs have increased interest in non-probability data sources such as opt-in web panels, social media, scanner data, mobile phone data, and voluntary registers. Since these sources lack known selection mechanisms, standard design-based inference methods cannot be directly applied. That is why we have developed the `nonprobsvy` package in the R language (version 0.2.1; available on CRAN).

The `nonprobsvy` package has several advantages over packages currently available in R or Python. In particular, the novelty and our contribution can be summarised as follows:

- the package implements state-of-the-art methods recently proposed in the literature, along with valid statistical inference procedures (i.e. analytical and boostrap variance estimators),
- the package implements various approaches, such as calibrated inverse probability weighting, mass imputation, and doubly robust estimators, with our contributions that extend existing literature,
- the package supports the functions included in the `survey` package to account for the design of the probability sample (if is available).

We provide a user-friendly API that mimics `glm`, `svydesign` and other functions known in R, together with the main function to specify the approach and estimators.

The package has been developed since 2022 and the full history can be found at the GitHub repository `https://github.com/ncn-foreigners/nonprobsvy`. The package has been cited multiple times (cf. `https://scholar.google.com/scholar?q=nonprobsvy`), including in the review paper by *Cobo et al. (2024). Software review for inference with non-probability surveys. The Survey Statistician, 90, 40-47.* Our package is also included in *West et al. (2025). Applied Survey Data Analysis (3rd ed.). Taylor & Francis.*

As far as we know, the `nonprobsvy` is the only software (open-access or commercial) that offers such functionalities. That is why we believe that the paper and software will be of interest to the readership of the Journal of Statistical Software.

Thank you for your consideration of this manuscript.

Sincerely,

Maciej Beręsewicz, Łukasz Chrostowski & Piotr Chlebicki

Replies:

- The article "nonprobsvy – An R package for modern methods for non-probability surveys" presents an R package implementing various methods for inference based on non-probability samples.
  The submission clearly falls within the aims and scope of Journal of Statistical Software. However, we found some issues that we would like to ask the authors to fix before we send this into full review. Please have a look at the issues raised below and resubmit a revised version along with a point-by-point answer.
  Detailed comments: the discussion on existing software implementations seems to only cover the R cosmos in detail. For JSS, it would be important to also discuss alternative implementations available for other statistical software packages and environments.
- Reply: in the review we focus on the R and Python libraries. In the revised version we have added references to general references for Stata and SAS software and removed the GJRM package as it implements a specific method that is not directly applicable to non-probability samples and instead of `rstanarm` we cite the Stan language. Please note that we have focused on software that can infer target parameters based on non-probability samples or data integration approaches.

- It is unclear why the authors have chosen to duplicate certain of their functions instead of implementing specific arguments to select the type of methods to use, e.g.:
  `method_glm`
  `method_nn`
  `method_npar`
  `method_pmm`
  This reduces the modularity of the implementation and the ability to extend the package functionality.
- Reply: We appreciate the editor's feedback on our package design. Our current implementation of separate methods was intentionally designed to enhance user experience and align with existing R packages. For example, the `maxLik` package uses a similar approach with methods like `maxBFGS` and `maxAdam`, and `WeightIt` also employs an internal `method_*` approach for its implemented methods.
  We initially chose to create distinct functions for each method (generalized linear models, nearest neighbours, non-parametric approaches, and predictive mean matching) to allow users to easily select and work with specific estimators they are most familiar with. This approach provides clarity and transparency in method selection.
  While we acknowledge that a more modular design with a single function and method-selection arguments could potentially improve the package's extensibility, reimplementing it at this stage would require substantial developmental effort.

- `help(package = "nonprobsvy")` shows that titles are not in title style.
- Reply: corrected.

- The main class has associated methods
  ```
  > methods(class = "nonprob")
  [1] anova check_balance confint nobs plot pop_size print  [8] summary update weights
  see '?methods' for accessing help and source code
  ```
  This raises the question how one would be easily able to extract the point estimates. This also relates to the issue that multiple times in the replication script the internal structure of objects is exposed, e.g.,
  `mi_est1_sel$outcome$single_shift`
  This is not a recommended practice. Instead, we recommend specific methods for accessing objects' internal structure.
- Reply: we have added two methods: `extract` to extract what is in `object$output` and `object$confidence_interval` and `coef` to extract coefficients of outcome or selection models. See `?extract` and `?coef` for examples.

- We see
  `plot(dr_est1)`

  ```
  Error in plot.nonprob(dr_est1) :

  We do not provide tools for visual assessment of the results.
  If you are interested in covariate balance plots, we recommend using the `cobalt` package.
  If you are interested in evaluation of the models (e.g. IPW, MI), we recommend using
  base R functions or the `modelsummary` package.
  ```

  It would seem that providing at least some plot would be appreciated by users and help them use and adopt the package.
- Reply: we have added a generic **plot** method which compares the naive (uncorrected) with the adjusted estimates along with confidence intervals.

- In the package we see:
  `t(X_nons * model_fitted$family$mu.eta(eta_nons)) %*% X_nons`
  `solve(-hess)`
  Usually, using solve to inverse matrices is inefficient (exploiting the specific structure of the matrix is usually beneficial when possible and we believe that it should be the case here). In addition, (t)crossprod should also be preferred over `'%*%'` for matrix multiplication.
- Reply: where possible we have replaced `'%*%'` with `(t)crossprod` and instead of solve we use `chol2inv(chol(matrix))`.

- Running
  `example(nonprob)`

  ```
  nonprb> library(sampling)
  Error in library(sampling) : there is no package called ‘sampling’
  ```

  Note that suggested packages should only be conditionally used. See Writing R Extensions at `https://cran.r-project.org/doc/manuals/r-devel/R-exts.html#Suggested-packages`.
- Reply: We have removed the `sampling` package completely, as we only use the sampling package in the examples of the `nonprob` function.