Journal of Statistical Software

MMMMMM YYYY, Volume VV, Issue II. *doi: 10.18637/jss.v000.i00*

nonprobsvy – An R package for modern methods for non-probability surveys

Łukasz Chrostowski
Adam Mickiewicz University

Piotr Chlebicki
Stockholm University

Maciej Beręsewicz
Poznań University of Economics and Business

Abstract

The abstract of the article.

Keywords: data integration, doubly robust estimation, propensity score estimation, mass imputation, R.

1. Introduction

With the availability of large sets of administrative data, voluntary internet panels, social media and big data, inference with non-probability samples is being heavily studied in the statistical literature Beaumont (2020), Elliott and Valliant (2017), Beręsewicz (2017), Citro (2014). Because of their non-statistical character and unknown sampling mechanism, these sources cannot be used directly for estimating population characteristics.

Several inference approaches have been proposed in the literature with respect to data from non-probability samples, which either involve data integration with population level data or probability samples from the same population (for recent review see Wu (2022)).

Although probability samples are still the most popular standard among statisticians, the cost of obtaining them, in terms of time or capital, motivates the use of non-probability

Table 1: Probability and non-probability samples

Factor	Probability sample	Non-probability sample
Selection	Sampling design	Auto-selection
Coverage	Typically good	Certain groups are excluded
Bias	Typically smaller	Large or very large
Variance	Typically larger	Small, or very small
Cost	Large or very large	Typically small

samples, which have to overcome other challenges. The first is that such samples generally do not represent the whole population, as can be said of probability samples. Another problem is the lack, or rather the ignorance, of the mechanism for selecting individuals for this type of sample, which does not allow the substantial use of existing statistical methods. For this reason, many different techniques have been proposed in the literature for integrating data in order to infer from available sources of different structural character.

It should be noted that there are several packages that allow the correction of selection bias in nonprobability samples, such as Marra and Radice (2023), Luis Castro Martín and del Mar Rueda (2020) or even Tillé and Matei (2021). However, these packages do not implement state-of-the-art approaches recently proposed in the literature: Chen, Li, and Wu (2020), Yang, Kim, and Song (2020), Wu (2022) nor do they use the survey package Lumley (2004) for inference.

This paper describes the nonprobsvy package for inference with non–probability samples, available from the Comprehensive R Archive Network (CRAN) at CRAN.R-project. Development version of the package can be also found at github.

Table 1 shows the basic characteristics of each of the samples described. In particular, what are the advantages and disadvantages of each type of sample with respect to population coverage, bias, variance, costs, and the selection mechanism for observations into the samples.

2. Methods for non-probability samples R code

2.1. Basic setup

Let $U = \{1, ..., N\}$ denote the target population consisting of N labelled units. Each unit i has an associated vector of auxiliary variables $\boldsymbol{x}_i$ (a realisation of the random vector $\boldsymbol{X}_i$ in the super-population) and the study variable y_i (a realisation of the random variable Y_i in the super-population). Let $\{(y_i, \boldsymbol{x}_i), i \in S_A\}$ be a dataset of a non-probability sample of size n_A and let $\{(\boldsymbol{x}_i, \pi_i), i \in S_B\}$ be a dataset of a probability sample of size n_B, where only information about variables $\boldsymbol{X}$ and inclusion probabilities π (which in the super population model are also considered to be random variables) are available. Let

Table 2: Two Sample Setting

Sample		Auxiliary Variables $\boldsymbol{X}$	Target Variable Y	Design (d) or Calibration (w) Weights
S_A (non-probability)	1	✓	✓	?
	...	✓	✓	?
	n_A	✓	✓	?
S_B (probability)	$n_A + 1$	✓	?	✓
	...	✓	?	✓
	$n_A + n_B$	✓	?	✓

δ be an indicator of inclusion into non-probability sample. Each unit in the sample S_B has been assigned a~design-based weight given by $d_i = 1/\pi_i$. The setting is summarised in Table ..

The goal is to estimate a~finite population mean $\mu_y = \frac{1}{N}\sum_{i=1}^{N} y_i$ of the target variable Y. As values of y_i are not observed in the probability sample, it cannot be used to estimate the target quantity. Instead, one could try combining the non-probability and probability samples to estimate μ_y. In this paper we do not consider modifications for the possibly occurring overlap. The above description of the data is presented in a more concise form in Table~2.

2.2. Mass Imputation estimators

Imputation refers to the process of replacing missing or incomplete data with substituted values. The goal of imputation is to allow for more complete data analysis, as many statistical methods require complete datasets. Common imputation techniques include:

- Mean imputation: where missing values are replaced by the mean of the observed data for that variable.
- Median imputation: where the median value of the observed data is used.
- Regression imputation: where missing values are estimated based on a regression model built from other available data.

Imputation helps prevent data bias and maintains dataset size, ensuring that missing data points do not skew analysis results. It is particularly useful when data is missing at random or when only a small portion of the data is missing.

Mass imputation is the application of imputation techniques to entire datasets where many observations have missing values for the given variable. Kim, Park, Chen, and Wu (2021), Yang, Kim, and Hwang (2021), Chlebicki, Chrostowski, and Beręsewicz (2024) propose the following imputation strategies as:

- Model based apprach (GLM),
- Nearest neigbour imputation (NN),

- Predictive mean mathing (PMM).

Mass imputation is particularly useful in large datasets where missing data can be widespread, and it seeks to preserve the relationships between variables, thus improving the overall integrity of the data.

By assumptions (Table 2), we do not know the value of the dependent variable Y for the units in the probability sample. In this case, the method will be to impute the values of the explanatory variable for all units in the probability sample. We therefore treat the non-probability sample as a training set that is used to build the imputation model. In this subsection, we distinguish three main methods of mass imputation based on linear models and the k-nearest neighbours algorithm. Other popular methods for estimating the variable Y from the variable $\boldsymbol{X}$ can also be considered, e.g. machine learning models such as random forests or neural networks.

We can obtain an estimate of the population mean based on known design weights and an imputation model for units from the probability sample:

$$\hat{\mu}_{MI} = \frac{1}{\hat{N}^{\mathrm{B}}} \sum_{i \in S_{\mathrm{B}}} d_i^{\mathrm{B}} \hat{y}, \tag{1}$$

$\hat{N}^{\mathrm{B}} = \sum_{i \in S_B} d_i^B$ and $\hat{y}$ is the estimated value of y for units from probability samples based on mass imputation model.

This estimator can be understood as a version of the Horvitz–Thompson estimator, which are used to estimate mean or total values in the population (based on probability sampling and inclusion probabilities). The only difference is that in our case, instead of the known values of the Y variable, we use its estimated equivalents.

2.3. Inverse Probability Weighting estimators

The main disadvantage of non-probability sampling is the unknown selection mechanism for a unit to be included in the sample. This is why we talk about the so-called "biased sample' ' problem. The inverse probability approach is based on the assumption that a reference probability sample is available and therefore we can estimate the propensity score of the selection mechanism. In recent years, a number of articles have addressed this issue. Chen *et al.* (2020) propose maximum likelihood estimation approach for estimating propensity scores for selection mechanism. Wu (2022) present the approach based on generalized estimating equations, this method is also mentioned in Yang *et al.* (2020). On the other hand calibration approach for quantiles was explained Beręsewicz and Szymkowiak (2024) and Sant'Anna, Song, and Xu (2022) present the approach based on maximize the covariate distribution balance among different treatment groups.

In the formal framework, let us introduce the following assumptions for propensity score model, which will imply a number of properties derived in the thesis.

(A1) The selection indicator R_i^A and explanatory variable y_i are independent.

(A2) All units have a so-called non-probability sample propensity score, which is non-zero, i.e. $\pi_i^{\mathrm{A}} > 0$, where $\pi_i^{\mathrm{A}} = P_q\left(R_i^A = 1 \mid \boldsymbol{x}_i, y_i\right)$, where q refers to the model for the selection mechanism for the non-probability sample (propensity score model).

(A3) Indicator variables R_i^A and R_j^A are independent with $i \neq j$.

The estimated propensity score is used to construct an inverse probability weighting estimator of the population mean of the form

$$\hat{\mu}_{IPW} = \frac{1}{\hat{N}^A} \sum_{i \in S_A} \frac{y_i}{\hat{\pi}_i^A}. \tag{2}$$

where $\hat{N}^A = \sum_{i \in S_A} \hat{d}_i^A = \sum_{i \in S_A} \frac{1}{\hat{\pi}_i^A}$.

2.4. Doubly Robust estimators

The inverse probability weighting and mass imputation estimators are sensible on misspecified models for propensity score and outcome variable respectively. For this purpose so called doubly-robust methods, which take into account these problems, are presented.

The proposed estimation procedure addresses the challenge of combining data from non-probability and probability survey samples. Traditional semiparametric models, often applied to such problems, are not directly usable in this context due to the distinct nature of the two samples. Instead, a joint randomization framework is employed, integrating semiparametric models for propensity scores with outcome regression for the nonprobability sample and design-based inference from the probability sample. This framework leads to a doubly robust (DR) estimation approach, which is effective in the presence of model misspecifications.

Inverse Probability Weighted (IPW) estimators are sensitive to misspecified propensity score models, particularly when propensity scores are very small. To improve robustness and efficiency, the doubly robust method incorporates a prediction model for the response variable. Moreover, even if one of the models is misspecified, the DR estimator remains consistent, showcasing the “double robustness’ ’ property.

Joint Randomization Approach

The joint randomization approach combines two processes: the selection mechanism of a non-probability sample, modeled by propensity scores, and the design-based inference from a probability sample.

The response y_i is predicted using a regression model $m(\boldsymbol{x}_i, \boldsymbol{\beta})$ (or NN/PMM methods), where $\boldsymbol{\beta}$ is estimated from the non-probability sample. With known design weights d_i^B for $i \in S_B$ we can define the DR estimator as

$$\hat{\mu}_{\mathrm{DR}} = \frac{1}{\hat{N}^{\mathrm{A}}} \sum_{i \in S_{\mathrm{A}}} d_i^{\mathrm{A}} \left\{y_i - m\left(\boldsymbol{x}_i, \hat{\boldsymbol{\beta}}\right)\right\} + \frac{1}{\hat{N}^{\mathrm{B}}} \sum_{i \in S_{\mathrm{B}}} d_i^{\mathrm{B}} m\left(\boldsymbol{x}_i, \hat{\boldsymbol{\beta}}\right), \tag{3}$$

where $d_i^A = \pi\left(\boldsymbol{x}_i, \boldsymbol{\theta}\right)^{-1}, \hat{N}^A = \sum_{i \in S_A} d_i^A$ and $\hat{N}^B = \sum_{i \in S_B} d_i^B$.

It remains consistent if either the propensity score model $\pi(\boldsymbol{x}_i, \boldsymbol{\theta})$ or the outcome regression model $m(\boldsymbol{x}_i, \boldsymbol{\beta})$ is correctly specified.

The joint randomization approach ensures robustness by accounting for randomness in both the non-probability sample through $\pi(\boldsymbol{x}_i, \boldsymbol{\theta})$ and the probability sample through design-based inference.

Minimization of the bias for doubly robust methods

By reducing the variance of the estimators, for example by variable selection, we cannot control the bias of the estimator, which may increase. Therefore, according to Yang *et al.* (2020), the idea is to determine the equations leading to the estimation of the $\boldsymbol{\beta}$ and $\boldsymbol{\theta}$ parameters based on the bias of the population mean estimator. In contrast to the joint randomization approach, this method allows for the estimation of the parameters $\boldsymbol{\beta}$ and $\boldsymbol{\theta}$ in a single step, rather than in two separate steps.

We will first present the bias of the doubly robust estimator and then, using optimisation techniques, discuss the equations leading to its minimization. Thus we have

$$\begin{aligned} \operatorname{bias}\left(\hat{\mu}_{DR}\right) &= \mid \hat{\mu}_{DR} - \mu \mid \\ &= \frac{1}{N} \sum_{i=1}^N \left\{\frac{R_i^A}{\pi_i^A\left(\boldsymbol{x}_i^{\mathrm{T}} \boldsymbol{\theta}\right)} - 1\right\}\left\{y_i - m\left(\boldsymbol{x}_i^{\mathrm{T}} \boldsymbol{\beta}\right)\right\} \\ &+ \frac{1}{N} \sum_{i=1}^N \left(R_i^B d_i^B - 1\right) m\left(\boldsymbol{x}_i^{\mathrm{T}} \boldsymbol{\beta}\right) \end{aligned} \tag{4}$$

To minimize $\operatorname{bias}\left(\hat{\mu}_{DR}\right)^2$ let us calculate the gradient of the square of the bias at $(\boldsymbol{\beta}, \boldsymbol{\theta})$. We then have

$$\frac{\partial \operatorname{bias}\left(\hat{\mu}_{DR}\right)^2}{\partial\left(\boldsymbol{\beta}^{\mathrm{T}}, \boldsymbol{\theta}^{\mathrm{T}}\right)^{\mathrm{T}}} = 2 \operatorname{bias}\left(\hat{\mu}_{DR}\right) J(\theta, \beta),$$

where

$$J(\theta, \beta) = \left(\begin{array}{c} J_1(\theta, \beta) \\ J_2(\theta, \beta) \end{array}\right) = \left(\begin{array}{c} \sum_{i=1}^N R_i^A\left\{\frac{1}{\pi\left(\boldsymbol{x}_i, \boldsymbol{\theta}\right)} - 1\right\}\left\{y_i - m\left(\boldsymbol{x}_i, \boldsymbol{\beta}\right)\right\} \boldsymbol{x}_i \\ \sum_{i=1}^N \frac{R_i^A}{\pi\left(\boldsymbol{x}_i, \boldsymbol{\theta}\right)} \frac{\partial m\left(\boldsymbol{x}_i, \boldsymbol{\beta}\right)}{\partial \boldsymbol{\beta}} - \sum_{i \in S_{\mathrm{B}}} d_i^{\mathrm{B}} \frac{\partial m\left(\boldsymbol{x}_i, \boldsymbol{\beta}\right)}{\partial \boldsymbol{\beta}} \end{array}\right),$$

which leads to the problem of solving the following system of equations

$$\left(\begin{array}{c} \sum_{i=1}^N R_i^A\left\{\frac{1}{\pi\left(\boldsymbol{x}_i, \boldsymbol{\theta}\right)} - 1\right\}\left\{y_i - m\left(\boldsymbol{x}_i, \boldsymbol{\beta}\right)\right\} \boldsymbol{x}_i \\ \sum_{i=1}^N \frac{R_i^A}{\pi\left(\boldsymbol{x}_i, \boldsymbol{\theta}\right)} \frac{\partial m\left(\boldsymbol{x}_i, \boldsymbol{\beta}\right)}{\partial \boldsymbol{\beta}} - \sum_{i \in S_{\mathrm{B}}} d_i^{\mathrm{B}} \frac{\partial m\left(\boldsymbol{x}_i, \boldsymbol{\beta}\right)}{\partial \boldsymbol{\beta}} \end{array}\right) = \mathbf{0}, \tag{5}$$

which can be solved using Newton–Raphson optimization method.

3. Package contents and implementation

All of the methods described in this paper have been implemented in the R package nonprobsvy Łukasz Chrostowski and Beręsewicz (2024). In this chapter, we will show you how to use the main nonprob function of the package and what its main features are. The package has been written to be as compatible as possible with the survey package for probablistic inference. Namely, the first step to use the nonprobsvy package is to define an object using the svydesign function that stores the probability sample data.frame and other objects, such as design weights. This is a negligible step if, instead of the probability sample, we have access to the values of the vector of sums of variables in the population. It is also worth mentioning that in order to speed up the calculations in the case of variable selection, part of the package, or more precisely the whole variable selection algorithm, was written in C++ using the Rcpp (Eddelbuettel, Francois, Allaire, Ushey, Kou, Russell, Ucar, Bates, and Chambers (2024)) package, which allows the C++ code to be called in the R environment. Moreover, the package is supported by other R packages such as foreach Folashade Daniel Hong Ooi and Weston (2023) (looping construct), maxLik Henningsen and Toomet (2023) (maximum likelihood estimation), Matrix Bates, Maechler *et al.* (2023) (matrix operations), MASS Ripley, Venables *et al.* (2023) (statistical functions and datasets), ncvreg Breheny and Huang (2023) (regularization methods), mathjaxr Epskamp (2023) (rendering equations in documentation), nleqslv Groemping (2023) (solving nonlinear equations), and doParallel Steve Weston and Tenenbaum (2022) (parallel computing).

3.1. Usage

```
R> nonprob(
+   data,
+   selection = NULL,
+   outcome = NULL,
+   target = NULL,
+   svydesign = NULL,
+   pop_totals = NULL,
+   pop_means = NULL,
+   pop_size = NULL,
+   method_selection = c("logit", "cloglog", "probit"),
+   method_outcome = c("glm", "nn", "pmm"),
+   family_outcome = c("gaussian", "binomial", "poisson"),
+   subset = NULL,
+   strata = NULL,
+   weights = NULL,
+   na_action = NULL,
+   control_selection = controlSel(),
+   control_outcome = controlOut(),
+   control_inference = controlInf(),
+   start_selection = NULL,
+   start_outcome = NULL,
+   verbose = FALSE,
+   x = TRUE,
+   y = TRUE,
+   se = TRUE,
+   ...
+ )
```

3.2. Arguments

Below is the definition of most of the arguments we can pass to the function. These are described in more detail in the documentation on the CRAN platform.

Argument	**Description**
data	Data frame with data from the non-probability sample.
selection	Formula for the selection (propensity) equation.
outcome	Formula for the outcome equation.
target	Formula with target variables.
svydesign	Optional svydesign object containing probability sample and design weights.
pop_totals	Optional named vector with population totals of the covariates.
pop_means	Optional named vector with population means of the covariates.
pop_size	Optional double with population size.
method_selection	Character string specifying the method for propensity score estimation (e.g., "logit").
method_outcome	Character string specifying the method for response variable estimation (e.g., "glm").
family_outcome	Character string describing the error distribution and link function to be used in the model (e.g., "gaussian").
subset	Optional vector specifying a subset of observations to be used in the fitting process.
strata	Optional vector specifying strata.
weights	Optional vector of prior weights to be used in the fitting process.
na_action	Function indicating what should happen when the data contain NAs.
control_selection	List indicating parameters to use in fitting selection model for propensity scores.
control_outcome	List indicating parameters to use in fitting model for outcome variable.

`control_inference` List indicating parameters to use in inference based on probability and non-probability samples.

`start_selection` Optional vector with starting values for the parameters of the selection equation.

`start_outcome` Optional vector with starting values for the parameters of the outcome equation.

`verbose` Logical value indicating if verbose output should be printed.

`x` Logical value indicating whether to return the model matrix of covariates as part of the output.

`y` Logical value indicating whether to return the vector of outcome variable as part of the output.

`se` Logical value indicating whether to calculate and return the standard error of the estimated mean.

`...` Additional optional arguments.

```
R> nonprob(
+   outcome = y ~ x1 + x2 + ... + xk,
+   data = nonprob,
+   svydesign = prob,
+   method_outcome = "glm",
+   family_outcome = "gaussian"
+ )

R> nonprob(
+   selection =  ~ x1 + x2 + ... + xk,
+   target = ~ y,
+   data = nonprob,
+   svydesign = prob,
+   method_selection = "logit"
+ )

R> nonprob(
+   selection = ~ x1 + x2 + ... + xk,
+   outcome = y ~ x1 + x2 + ... + xk,
+   data = nonprob,
+   svydesign = prob,
+   method_outcome = "glm",
+   family_outcome = "gaussian",
+   method_selection = "logit"
+ )

R> nonprob(
+   selection = ~ x1 + x2 + ... + xk,
+   outcome = y ~ x1 + x2 + ... + xk,
+   data = nonprob,
+   svydesign = prob,
+   method_outcome = "glm",
+   family_outcome = "gaussian",
+   method_selection = "logit",
+   control_selection = controlSel(penalty = "SCAD"),
+   control_outcome = controlSel(penalty = "MCP"),
+   control_inference = controlInf(vars_selection = TRUE),
+   verbose = TRUE
+ )
```

```
R> nonprob(
+   selection = ~ x1 + x2 + ... + xk,
+   outcome = y ~ x1 + x2 + ... + xk,
+   data = nonprob,
+   svydesign = prob,
+   method_outcome = "glm",
+   family_outcome = "gaussian",
+   method_selection = "logit",
+   control_selection = controlSel(est_method_sel = "gee", h = 2))
```

```
R> nonprob(
+   selection = ~ x1 + x2 + ... + xk,
+   outcome = y ~ x1 + x2 + ... + xk,
+   data = nonprob,
+   svydesign = prob,
+   method_outcome = "nn",
+   family_outcome = "gaussian",
+   method_selection = "logit",
+   control_outcome = controlOut(k = 3)
+ )
```

```
R> nonprob(
+   selection = ~ x1 + x2 + ... + xk,
+   outcome = y ~ x1 + x2 + ... + xk,
+   data = nonprob,
+   svydesign = prob,
+   method_outcome = "glm",
+   family_outcome = "gaussian",
+   method_selection = "logit",
+   control_selection = controlSel(penalty = "SCAD"),
+   control_inference = controlInf(vars_selection = TRUE,
+                                  bias_correction = TRUE,
+                                  var_method = "bootstrap"),
+   verbose = TRUE
+ )
```

```
R> Call:
+ nonprob(data = nonprob_df, selection = ~x1 + x2 + x3 + x4, outcome = y30 ~
+     x1 + x2 + x3 + x4, svydesign = svyprob, method_selection = "logit")
+
+ -------------------------
+ Estimated population mean: 9.374 with overall std.err of: 0.3955
+ And std.err for nonprobability and probability samples being respectively:
+ 0.364 and 0.1546
+
+ 95% Confidence interval for population mean:
+     lower_bound upper_bound
+ y30    8.598809    10.14916
+
+ Based on: Doubly-Robust method
+ For a population of estimate size: 20200.83
+ Obtained on a nonprobability sample of size: 950
+ With an auxiliary probability sample of size: 1001
+ -------------------------
+
+ Regression coefficients:
+ -----------------------
+ For glm regression on outcome variable:
+             Estimate Std. Error z value  P(>|z|)
+ (Intercept) -0.44155    1.05815  -0.417 0.676469
+ x1           1.20976    0.72852   1.661 0.096802 .
+ x2           1.50153    0.44714   3.358 0.000785 ***
+ x3           1.35890    0.28206   4.818 1.45e-06 ***
+ x4           1.17264    0.08029  14.606  < 2e-16 ***
+ ---
+ Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1
+
+ -----------------------
+ For glm regression on selection variable:
+              Estimate Std. Error z value  P(>|z|)
+ (Intercept) -4.480634   0.113868 -39.349  < 2e-16 ***
+ x1          -0.028191   0.074889  -0.376    0.707
+ x2           0.277772   0.044721   6.211 5.26e-10 ***
+ x3           0.148772   0.029746   5.001 5.69e-07 ***
+ x4           0.172832   0.008622  20.046  < 2e-16 ***
+ -------------------------
+
+ Weights:
+    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
+   1.143  11.180  19.502  21.264  28.089  79.119
+ -------------------------
+
+ Covariate balance:
+ (Intercept)          x1          x2          x3          x4
+    72.18047  -425.80885  -584.02397  -351.61222  1508.57831
+ -------------------------
+
+ Residuals:
+     Min.   1st Qu.   Median     Mean  3rd Qu.     Max.
+ -0.31329 -0.04205 -0.01799  0.42335  0.94734  0.98736
+
+ AIC: 7255.979
+ BIC: 7283.86
+ Log-Likelihood: -3622.99 on 1946 Degrees of freedom
```

4. Practical examples

Summary

Bates D, Maechler M, *et al.* (2023). *Matrix: Sparse and Dense Matrix Classes and Methods*. R package version 1.6-1.1, URL https://CRAN.R-project.org/package=Matrix.

Beaumont JF (2020). "Are probability surveys bound to disappear for the production of official statistics." *Survey Methodology*, **46**(1), 1–28.

Beręsewicz M (2017). "A two-step procedure to measure representativeness of internet data sources." *International Statistical Review*, **85**(3), 473–493.

Beręsewicz M, Szymkowiak M (2024). "Inference for non-probability samples using the calibration approach for quantiles." https://arxiv.org/abs/2403.09726. 2403.09726.

Breheny P, Huang J (2023). *ncvreg: Regularization Paths for SCAD, MCP, and Elastic Net*. R package version 3.14-0, URL https://CRAN.R-project.org/package=ncvreg.

Chen Y, Li P, Wu C (2020). "Doubly robust inference with nonprobability survey samples." *Journal of the American Statistical Association*, **115**(532), 2011–2021.

Chlebicki P, Chrostowski Ł, Beręsewicz M (2024). "Data integration of non-probability and probability samples with predictive mean matching."

Citro CF (2014). "From multiple modes for surveys to multiple data sources for estimates." *Survey Methodology*, **40**(2), 137–162.

Eddelbuettel D, Francois R, Allaire J, Ushey K, Kou Q, Russell N, Ucar I, Bates D, Chambers J (2024). *Rcpp: Seamless R and C++ Integration*. R package version 1.0.13, URL `https://CRAN.R-project.org/package=Rcpp`.

Elliott MR, Valliant R (2017). "Inference for Nonprobability Samples." *Statistical Science*, **32**(2). ISSN 0883-4237. `doi:10.1214/16-STS598`. URL `https://projecteuclid.org/journals/statistical-science/volume-32/issue-2/Inference-for-Nonprobability-Samples/10.1214/16-STS598.full`.

Epskamp S (2023). *mathjaxr: Using MathJax in Rd Files for Dynamic Rendering of Equations*. R package version 1.6-0, URL `https://CRAN.R-project.org/package=mathjaxr`.

Folashade Daniel Hong Ooi RC, Weston S (2023). *foreach: Provides Foreach Looping Construct for R*. R package version 1.5.2, URL `https://CRAN.R-project.org/package=foreach`.

Groemping U (2023). *nleqslv: Solve Systems of Nonlinear Equations*. R package version 3.3.3, URL `https://CRAN.R-project.org/package=nleqslv`.

Henningsen A, Toomet O (2023). *maxLik: Maximum Likelihood Estimation and Related Tools*. R package version 1.8-5, URL `https://CRAN.R-project.org/package=maxLik`.

Kim JK, Park S, Chen Y, Wu C (2021). "Combining Non-Probability and Probability Survey Samples Through Mass Imputation." *Journal of the Royal Statistical Society Series A: Statistics in Society*, **184**(3), 941–963. ISSN 0964-1998, 1467-985X. `doi:10.1111/rssa.12696`. URL `https://academic.oup.com/jrsssa/article/184/3/941/7068406`.

Luis Castro Martín RFG, del Mar Rueda M (2020). *NonProbEst: Estimation in Nonprobability Sampling*.

Lumley T (2004). "survey R package."

Marra G, Radice R (2023). *GJRM: Generalised Joint Regression Modelling*.

Ripley B, Venables W, *et al.* (2023). *MASS: Support Functions and Datasets for Venables and Ripley's MASS*. R package version 7.3-60, URL `https://CRAN.R-project.org/package=MASS`.

Sant'Anna PHC, Song X, Xu Q (2022). "Covariate Distribution Balance via Propensity Scores." *Journal of Applied Econometrics*, **37**(6), 1093–1120.

Steve Weston Folashade Daniel SW, Tenenbaum D (2022). *doParallel: Foreach Parallel Adaptor for the 'parallel' Package*. R package version 1.0.17, URL https://CRAN.R-project.org/package=doParallel.

Tillé Y, Matei A (2021). *sampling: Survey Sampling*. R package version 2.9, URL https://CRAN.R-project.org/package=sampling.

Wu C (2022). "Statistical inference with non-probability survey samples." *Survey Methodology*, **48**, 283–311.

Yang S, Kim JK, Hwang Y (2021). "Integration of data from probability surveys and big found data for finite population inference using mass imputation." *Survey Methodology*, **47**, 29–58.

Yang S, Kim JK, Song R (2020). "Doubly Robust Inference when Combining Probability and Non-Probability Samples with High Dimensional Data." *Journal of the Royal Statistical Society Series B: Statistical Methodology*, **82**(2), 445–465. ISSN 1369-7412, 1467-9868. doi:10.1111/rssb.12354. URL https://academic.oup.com/jrsssb/article/82/2/445/7056072.

Łukasz Chrostowski, Beręsewicz M (2024). *nonprobsvy: Package for Inference Based on Non-Probability Samples*. R package version 0.1.0, https://ncn-foreigners.github.io/nonprobsvy/, URL https://github.com/ncn-foreigners/nonprobsvy.

Affiliation:

Łukasz Chrostowski
Pearson
First line
Second line
E-mail: lukchr@st.amu.edu.pl
URL: https://posit.co

Piotr Chlebicki
Stockholm University
Department of Statistics and Mathematics,
Faculty of Biosciences,
Universitat Autònoma de Barcelona
E-mail: piotr.chlebicki@math.su.se

Maciej Beręsewicz
Statistical Office in Poznań
Department of Statistics,
Institute of Informatics and Electronic Economy,
Poznań University of Economics and Business
E-mail: maciej.beresewicz@ue.poznan.pl
URL: https://posit.co

Journal of Statistical Software http://www.jstatsoft.org/
published by the Foundation for Open Access Statistics http://www.foastat.org/
MMMMMM YYYY, Volume VV, Issue II *Submitted:* yyyy-mm-dd
doi:10.18637/jss.v000.i00 *Accepted:* yyyy-mm-dd