mathematics

MDPI

Article

Inference from Non-Probability Surveys with Statistical Matching and Propensity Score Adjustment Using Modern Prediction Techniques

Luis Castro-Martín, Maria del Mar Rueda * and Ramón Ferri-García

Department of Statistics and Operational Research, Faculty of Sciencies, University of Granada, 18071 Granada, Spain; luiscastro193@ugr.es (L.C.-M.); rferri@ugr.es (R.F.-G.)

* Correspondence: mrueda@ugr.es

Received: 7 May 2020; Accepted: 27 May 2020; Published: 1 June 2020

Abstract: Online surveys are increasingly common in social and health studies, as they provide fast and inexpensive results in comparison to traditional ones. However, these surveys often work with biased samples, as the data collection is often non-probabilistic because of the lack of internet coverage in certain population groups and the self-selection procedure that many online surveys rely on. Some procedures have been proposed to mitigate the bias, such as propensity score adjustment (PSA) and statistical matching. In PSA, propensity to participate in a nonprobability survey is estimated using a probability reference survey, and then used to obtain weighted estimates. In statistical matching, the nonprobability sample is used to train models to predict the values of the target variable, and the predictions of the models for the probability sample can be used to estimate population values. In this study, both methods are compared using three datasets to simulate pseudopopulations from which nonprobability and probability samples are drawn and used to estimate population parameters. In addition, the study compares the use of linear models and Machine Learning prediction algorithms in propensity estimation in PSA and predictive modeling in Statistical Matching. The results show that statistical matching outperforms PSA in terms of bias reduction and Root Mean Square Error (RMSE), and that simpler prediction models, such as linear and k-Nearest Neighbors, provide better outcomes than bagging algorithms.

Keywords: nonprobability surveys; machine learning; matching; propensity score adjustment; sampling

1. Introduction

Surveys are a fundamental tool for data collection in areas like social studies and health sciences. Probability sampling methods have been widely adopted by researchers in those areas, as well as by official statistics. The main reason is that it provides valid statistical inferences about large finite populations by using relatively small samples, based on a solid mathematical theory, with the right combination of a random sample design and an approximately design-unbiased estimator.

Over the last decade, new alternatives to survey sample data have became popular as data sources. Examples are big data and web surveys that have the potential of providing estimates in nearly real time, an easier data access, and lower data collection costs than traditional probability sampling [1]. Very often, the data-generating process of such sources is nonprobabilistic, given that the probability of being part of the sample is not known and/or is null for some groups of the target population, and, as a result, these methods produce nonprobability samples. There are serious issues on the use of nonprobability survey samples; the most relevant is that the data-generating process is unknown and may have serious coverage, nonresponse, and selection biases, which may not be ignorable and could

deeply affect estimates [2]. These biases tend to be more disruptive as the population size gets larger, regardless of the sample size [3].

In order to correct this selection bias produced by non-random selection mechanisms, some inference procedures are proposed in the literature. A first class of methods includes statistical models aiming to predict the non-sampled units of the population [4–6]. Specifying an appropriate super-population model capable of learning the variation of the target variables is important for this model-based method. For this approach, auxiliary features X must be available for each unit of the observed and the unobserved parts of the population. This situation is complicated in practice.

Some studies combine a nonprobability sample with a reference probability sample for constructing models for units in the latter or to adjust selection probabilities. The most important methods for this case are statistical matching and propensity score adjustment (PSA). There are many works studying the properties and performance of PSA [7–12], but there is not much of a bibliography that develops statistical matching in this context.

In this article, we apply machine learning prediction techniques to build statistical matching estimators and compare their performance with PSA estimators. Since there is not a sampling design that allows us to determine the main statistical properties (sampling distribution, expected value, variance, etc.) of random quantities calculated from the non-probability sample, we cannot include theoretical properties of the estimators obtained, but their behavior is studied through simulation studies that also include several propensity score techniques. Although PSA performance was compared with linear calibration in [10] and the combination of PSA with machine learning was already studied in [12], to the best of our knowledge, this is the first time that these methodologies are compared in practice and the first time that machine learning techniques are used for estimation with statistical matching from nonprobability samples.

The description of the conducted study is organized as follows: In Section 2, we introduce the notation and explain the estimation problem that can be solved with the aforementioned methods. In Sections 3 and 4, we describe the mathematical foundations of PSA and statistical matching, respectively, and their properties according to previous research. In Section 5, we briefly explain the ideas behind each of the algorithms tested in the study. In Section 6, we describe the data and the simulation study used to compare the performance of PSA and statistical matching, as well as the metrics used to measure it. Finally, in Sections 7 and 8, we show the results of the study and discuss some of their implications in the comparison between methods.

2. Background

Suppose that the finite population U consists of $i = 1, ..., N$ subjects. Let y be a survey variable and y_i be the y-value attached to the i-th unit,$i = 1, ..., N$.

Let s_v be a volunteer nonprobability sample of size n_v, obtained from $U_v \subset U$ observing the study variable y.

Without any auxiliary information, the population mean $\bar{Y}$ is usually estimated with the unweighted sample mean

$$\hat{\bar{Y}} = \sum_{k \in s_v} \frac{y_k}{n_v} \tag{1}$$

that produces biased estimates of the population mean. The size and direction of the bias depend on the proportion of the population with no chance of inclusion in the sample (coverage) and differences in the inclusion probabilities among the different members of the sample with a non-zero probability of taking part in the survey (selection) [2,13]. The selection bias cannot be estimated in practice for most survey variables of interest.

We consider the situation where there is a probability sample available and compare two inference methods to treat selection biases in a general framework. Let s_r be the reference probability sample selected under the sampling design (s_d, p_d) with π_i the first-order inclusion probability for the i-th individual. Let us assume that in s_r, we observe some other study variables that are common to

both samples, denoted by x. The available data are denoted by $\{(i, y_i, x_i), i \in s_v\}$, and $\{(i, x_i), i \in s_r\}$. We are interested in estimating a linear parameter $\theta_N = \sum_U a_i y_i$, where a_i are known constants. Examples include the population total $T_y = \sum_U y_i$, the population mean $\bar{Y}$, and the population proportion $p_A = \sum_U y_i / N$, where $y_i = 1$ if the unit i belongs to the interest group A, and 0 otherwise.

3. Propensity Score Adjustment

The most popular adjustment method in nonprobability settings is propensity score adjustment (PSA) or propensity weighting. This method, firstly developed by [14], was originally intended to correct the confounding bias in the experimental design context, and it is the most widely used method in practice [2,7–10,12,15–17]. In this approach, the propensity for an individual to participate in the volunteer survey is estimated by binning the data from both samples, s_r and s_v, together and training a machine learning model (usually logistic regression) on the variable δ, with $\delta_k = 1$ if $k \in s_v$ and $\delta_k = 0$ if $k \in s_r$. We assume that the selection mechanism of s_v is ignorable; this is:

$$P(\delta_k = i | y_k, \mathbf{x}_k) = P(\delta_k = i | \mathbf{x}_k), i = 0, 1; k \in s_v. \tag{2}$$

We also assume that the mechanism follows a parametric model:

$$P(\delta_k = 1 | y_k, \mathbf{x}_k) = p_k(\mathbf{x}) = \frac{1}{e^{-(\gamma^T \mathbf{x}_k)} + 1} \tag{3}$$

for some vector γ. We obtain the pseudo maximum likelihood of parameter γ and use the inverse of the estimated response propensity as weight for constructing the estimator [11]:

$$\hat{\theta}_{PSA1} = \sum_{k \in s_v} a_k y_k / \hat{p}_k(\mathbf{x}_k), \tag{4}$$

where $\hat{p}_k(\mathbf{x}_k)$ denotes the estimated response propensity for the individual $k \in s_v$. Alternative estimators can be constructed by slightly modifying the formula in (4) [18]:

$$\hat{\theta}_{PSA2} = \sum_{k \in s_v} (1 - \hat{p}_k(\mathbf{x}_k)) \, a_k y_k / \hat{p}_k(\mathbf{x}_k). \tag{5}$$

Other alternatives involve the stratification of propensities in a fixed number of groups, with the idea of grouping individuals with similar volunteering propensities. For instance, in [7,8], adjustment factors f_c are obtained for the cth strata of individuals:

$$f_c = \frac{\sum_{k \in s_r^c} d_k^r / \sum_{k \in s_r} d_k^r}{\sum_{j \in s_v^c} d_j^v / \sum_{j \in s_v} d_j^v}, \tag{6}$$

where s_r^c and s_v^c are individuals from the s_r and s_v sample respectively who belong to the cth group, while $d_k^r = 1/\pi_k$ and $d_j^v = 1/\hat{p}_j$ are the design weights for the kth individual of the reference sample and the jth individual of the volunteer sample, respectively. The final weights are:

$$w_j = f_c d_j^v = \frac{\sum_{k \in s_r^c} d_k^r / \sum_{k \in s_r} d_k^r}{\sum_{j \in s_v^c} d_j^v / \sum_{j \in s_v} d_j^v} d_j^v. \tag{7}$$

The weights are then used in the Horvitz–Thompson estimator:

$$\hat{\theta}_{PSA3} = \sum_{k \in s_v} w_k a_k y_k. \tag{8}$$

The approach used in [9] does not require the calculation of f_c. It only uses the average propensity within each group

$$\overline{\pi}_c = \sum_{k \in s_r^c \cup s_v^c} p_k(\mathbf{x}) / (n_r^c + n_v^c), \tag{9}$$

where n_r^c and n_v^c are the number of individuals from the reference and the volunteer sample, respectively, that belong to the cth group. The mean propensity for each member of the volunteer sample is used in the Horvitz–Thompson estimator:

$$\hat{\theta}_{PSA4} = \sum_c \sum_{k \in s_v^c} a_k y_k / \overline{\pi}_c. \tag{10}$$

PSA in nonprobability online surveys has been proven to be efficient if the selection mechanism is ignorable and the right covariates are used for modeling [7]. If some of these conditions do not apply, the use of PSA can induce biased estimates that would need further adjustments [9]. The combination of PSA and calibration has shown successful results in terms of bias removal [8,10].

Several machine learning models have been suggested as alternatives to logistic regression for the estimation of propensity scores in the experimental design context, with promising results. Ref. [19,20] examined the performance of various classification and regression trees (CART) for PSA in sample balancing. Other applications of machine learning algorithms in PSA involve their use in nonresponse adjustments; more precisely, they have been studied using Random Forests as propensity predictors [21]. Regarding the nonprobability sampling context covered in this study, [12] presented a simulation study using decision trees, k-Nearest Neighbors, Naive Bayes, Random Forests, and a Gradient Boosting Machine that support the view given in [6] about machine learning methods being used for removing selection bias in nonprobability samples. All of those algorithms, along with Discriminant Analysis and Model Averaged Neural Networks, will be used for propensity estimation in this study. Further details can be consulted in Section 5.

4. Statistical Matching

Statistical matching (also known as data fusion, data merging, or synthetic matching) is a model-based approach introduced by [22] and further developed by [23] for nonresponse in probability samples. The idea in this context is to model the relationship between y_k and x_k using the volunteer sample s_V in order to predict y_k for the reference sample.

Suppose that the finite population $\{(i, y_i, x_i), i \in U\}$ can be viewed as a random sample from the superpopulation model:

$$y_i = m(x_i) + e_i, i = 1, 2, ..., N, \tag{11}$$

where $m(x_i) = E_m(y_i|x_i)$ and the random vector $e = (e_1, ..., e_N)'$ is assumed to have zero mean.

Under the design-based approach, the usual estimator of a population's linear parameter is the Horvitz–Thompson estimator given by:

$$\hat{\theta}_{HT} = \sum_{k \in s_r} a_k y_k d_k \tag{12}$$

where $d_k = 1/\pi_k$ is the sampling weight of the unit k that is design-unbiased, consistent for θ, and asymptotically normally distributed under mild conditions [24]. This estimator cannot be calculated because y_k is not observed for the units $k \in s_r$; thus, we substitute y_i by the predicted values from the above model. Thus, the matching estimator is given by:

$$\hat{\theta}_{SM} = \sum_{s_r} a_k \hat{y}_k d_k, \tag{13}$$

where $\hat{y}_k$ is the predicted value of y_k.

The key is how to predict the values of y_k. Usually, the linear regression model is considered for estimation; $E_m(y_i|\mathbf{x}_i) = \mathbf{x}_i^T\beta$ is easy to implement in most of the existent statistical packages, but several drawbacks have to be considered. Parametric models require assumptions regarding variable selection, the functional form and distributions of variables, and specification of interactions. If any of these assumptions are incorrect, the bias reduction could be incomplete or nonexistent. Contrary to statistical modeling approaches that assume a data model with parameters estimated from the data, more advanced machine learning algorithms aim to extract the relationship between an outcome and predictor without an a priori data model. These methods have not been widely applied in the statistical matching literature. Now, we propose the use of machine learning methods as an alternative to linear regression modeling. The ML prediction methods considered in this article are described in the following section.

5. Prediction Modeling

5.1. Generalized Linear Models (GLM)

The most basic regression model consists of calculating coefficients, β, of linear regression based on input data. The coefficients that satisfy the optimality criteria based on minimizing the Ordinary Least Squares are estimated with the formula $\beta = (\mathbf{X}'\mathbf{X})^{-1}\mathbf{X}'\mathbf{Y}$. This method is only stable as long as $\mathbf{X}'\mathbf{X}$ is relatively close to a unit matrix [25]. Quite often, covariates suffer from multicollinearity. For those cases, ridge regression proposes an identity term to control instability: $\beta = (\mathbf{X}'\mathbf{X} + k\mathbf{I})^{-1}\mathbf{X}'\mathbf{Y}$, where $k \geq 0$ can be chosen arbitrarily or via parameter tuning. β can also be considered as the posterior mean of a prior normal distribution with zero mean and a variance of $\mathbf{I}\sigma^2/k$ [26]. From that point of view, Bayesian estimates can be obtained via Gibbs sampling.

Instead, the Least Absolute Shrinkage and Selection Operator (LASSO) regression [27] proposes using a penalty parameter, α, according to the following optimization problem:

$$\begin{array}{l} argmin \sum_{i=1}^{N}(y_i - \alpha - \sum_j \beta_j x_{ij})^2 \\ subject\ to \sum_j |\beta_j| \leq t. \end{array} \tag{14}$$

t is a hyperparameter that forces the shrinkage of the coefficients. In this case, coefficients are allowed to be equal to zero. Therefore, the main difference is that LASSO allows the optimization procedure to select variables, while ridge regression may produce very small coefficients for some cases without reaching zero. Alternatively, LASSO coefficients can be estimated considering the posterior mode of prior Laplace distributions. Bayesian estimates can then be calculated as described in [28]. Ridge and LASSO are both considered standard penalized regression models [29].

For PSA, these methods can be used for estimating the propensities. First, the target variable for the model is defined as $y_i = 1$ if $k \in s_v$ and $y_i = 0$ if $k \in s_r$. The pseudo maximum likelihood can then be optimized via logistic regression or any of its variants described above.

For statistical matching, the target variable for the model is the survey variable itself. Therefore, the model is trained with the volunteer sample and then used to obtain the estimated responses for the reference sample.

5.2. Discriminant Analysis

When the predicted variable is discrete, Discriminant Analysis can be used for classification of individuals. Let y be the dependent variable with K classes, π_k the probability of an individual of belonging to the kth class, $\mathbf{X}$ the matrix of covariates $n \times p$, and $f_k(\mathbf{x})$ the joint distribution of $\mathbf{x}$ conditioned to y taking the kth class. As described in [30], Linear Discriminant Analysis (LDA) assigns an individual the class that maximizes the probability:

$$P(y_i = k|\mathbf{x} = \mathbf{x}_i) = \frac{\pi_k f_k(\mathbf{x}_i)}{\sum_{j=1}^{K} \pi_j f_j(\mathbf{x}_i)}, k = 1, ..., K. \tag{15}$$

Assuming that $\mathbf{X}|y = k$ follows a multivariate Gaussian distribution $N_p(\mu_k, \Sigma)$, LDA works by assigning an instance the class for which the coefficient $\delta_k(\mathbf{x}_i)$ defined as

$$\delta_k(\mathbf{x}_i) = \mathbf{x}_i^T \Sigma^{-1} \mu_k - \frac{1}{2} \mu_k^T \Sigma^{-1} \mu_k + log(\pi_k) \tag{16}$$

is largest. Note that the decision depends on a linear combination of the multivariate Gaussian distribution parameters; hence, the classifier gets the name of Linear Discriminant Analysis. When used for PSA, $K = 2$ and, as a result, the outcome of LDA is the posterior probability obtained in (15) for the class $\delta = 1$.

LDA can provide good results; however, its simplicity can be a handicap in some cases where relationships between covariates and target are complex, and if the covariates are correlated, its performance gets worse [31]. For these reasons, alternatives considering smoothing, such as Penalized Discriminant Analysis (FDA) or Shrinkage Discriminant Analysis (SDA), can be used. The former expands the covariate matrix and applies penalization coefficients in the calculation of thresholds [32], while the latter performs a shrinkage of covariates similar to that performed in the ridge or LASSO models.

LDA is only suitable for classification and, therefore, it cannot be used for statistical matching when the survey variable is continuous. However, its probabilistic nature makes it appropriate for estimating propensities in PSA, as described above.

5.3. Decision Trees, Bagged Trees, and Random Forests

Decision trees sequentially split the input data via conditional clauses until they reach a terminal node, which assigns a specific class or value. This process results in the following estimation for the expectance $E_m(y_i|\mathbf{x}_i)$:

$$E_m(y_i|\mathbf{x}_i) = \begin{cases} \overline{y(s^{J_1})} & \{i \in s / \mathbf{x}_i \in J_1\} \\ \dots & \dots \\ \overline{y(s^{J_k})} & \{i \in s / \mathbf{x}_i \in J_k\}, \end{cases} \tag{17}$$

where $\overline{y(s^{J_i})}$ denotes the mean of y among the members of the sampled population, s, meeting the criteria of the i-th terminal node.

Bagged trees combine this approach with bagging [33]. Bagging averages the predictions of multiple weak classifiers (in this case, m unpruned trees). In order for them to complement each other, they are trained with a bootstrapped subsample of the complete dataset. Therefore:

$$E_m(y_i|\mathbf{x}_i) = \frac{\sum_{j=1}^{m} \phi_j(\mathbf{x}_i)}{m}, \; \phi_j(\mathbf{x}_i) = \begin{cases} \overline{y(s^{J_1^j})} & \{i \in s / \mathbf{x}_i \in J_1^j\} \\ \dots & \dots \\ \overline{y(s^{J_k^j})} & \{i \in s / \mathbf{x}_i \in J_k^j\}, \end{cases} \tag{18}$$

where $\overline{y(s^{J_i^j})}$ denotes the mean of y among the members of the sampled population, s, meeting the criteria of the i-th terminal node of the j-th tree. This technique is known to improve the accuracy of the predictions [34]. Alternatively, Random Forests can also be used for both regression and classification using weak classifiers [35]. In this algorithm, the input variables for each weak classifier are a random subset of all of the covariates, instead of taking the whole $\mathbf{x}_i$ vector as in bagged trees.

This approach is easy to apply for statistical matching. As usual, a model is trained using the volunteer sample in order to predict a response based on the covariates. Said model is then applied to the reference sample covariates. However, tree-based models are not good for estimating probabilities [36]. They can still be used for PSA, taking the proportion of weak classifiers that agree as the estimated propensity.

5.4. Gradient Boosting Machine

Gradient Boosting Machine (GBM) also works as an ensemble of weak classifiers. Boosting is an iterative process that trains subsequent models, giving more importance to the data for which previous models failed. This idea can be interpreted as an optimization problem [37], and, therefore, it is suitable for the gradient descent algorithm [38]. Then, the estimates for y are:

$$E_m(y_i|\mathbf{x}_i) = v^T J(\mathbf{x}_i), \tag{19}$$

where $J(\mathbf{x}_i)$ stands for a matrix of terminal nodes of m decision trees and v is a vector representing the weight of each tree. GBM has improved previous state-of-the-art models for some cases [39].

GBM can be used for PSA and statistical matching in the same way as the previous ensemble models considered.

5.5. k-Nearest Neighbors

k-Nearest Neighbors is "one of the most fundamental and simple classification methods" [40]. It does not need training. The algorithm simply averages the value of the target variable for the k individuals closest to the estimated individual (its k nearest neighbors), given a certain distance dependent on the covariates. This is:

$$E_m(y_i|\mathbf{x}_i) = \frac{\sum_{j \in s / d(\mathbf{x}_i, \mathbf{x}_j) \leq d(\mathbf{x}_i, \mathbf{x}_{(k)})} y_j}{k} \tag{20}$$

where $x_{(1)}$,...,$x_{(n-1)}$ are, respectively, the individuals closest to and furthest from x_i. Choosing the right k is important for the proper performance of the algorithm.

For classification, k-Nearest Neighbors would usually simply predict the most repeated label among its k-nearest neighbors. However, it can instead take into account the proportion in order to estimate probabilities. This idea can be applied for PSA, taking $y_i = 1$ if $k \in s_v$ and $y_i = 0$ if $k \in s_r$, as always. For statistical matching, k-Nearest Neighbors can also be used normally to predict the responses.

5.6. Naive Bayes

The Naive Bayes algorithm is a classifier (that is, it can only be used to predict discrete variables) based on the Bayes theorem. In this study, Naive Bayes has been used only for propensity estimation in PSA. In this case, the Bayes theorem can be used with the probabilities that the participants have of being part of the volunteer sample and the occurrence of a given vector for $\mathbf{X}$, that is, the values of the covariates for a given individual i.

$$\hat{p}_i(\mathbf{x}_i) = \frac{P(\delta_i = 1)P(\mathbf{X} = x_i|\delta_i = 1)}{P(\mathbf{X} = \mathbf{x}_i)}. \tag{21}$$

The Naive Bayes classifier is simple in its reasoning, but can provide precise results in PSA under certain conditions [12]. On the other hand, predictions from Naive Bayes can turn unstable when covariates with high cardinality (e.g., numerical variables) are present, as discrete domains are required for computation of probabilities [41].

As was the case with discriminant analysis, Naive Bayes works naturally with probabilities; therefore, it is suitable for estimating propensities in PSA, but not for statistical matching if the survey variable is continuous.

5.7. Neural Networks with Bayesian Regularization

Neural networks calculate the expectance of y_i as:

$$E_m(y_i|\mathbf{x}_i) = g\left(\sum_{k=1}^{L} v_k f_k(\cdot) + b\right), \tag{22}$$

where g and f_k stand for the activation functions, v_k are the weights of the k-th neuron, and b is the activation threshold [42]. The inputs follow an iterative process through one or more hidden layers until reaching the last layer, which produces the final output. The weights are initialized randomly and then optimized via gradient descent with the backpropagation algorithm [43].

Overfitting is an important problem for neural networks so prior distributions can be imposed to v_k weights as a regularization method. They are then optimized to maximize the posterior density or the likelihood, as described in [42]. Another option is bagging of neural networks, as explained in [44]. The same neural network model is fitted using different seeds, and the results are averaged to obtain the predictions. This approach is known as Model Averaged Neural Networks.

Neural networks have already been considered for superpopulation modeling [45]. They are the state of the art for many domains [46]; Bayesian neural networks in particular are "fairly robust with respect to the problems of overfitting and hyperparameter choice" [47].

Since they work as universal approximators [48], neural networks can be used for PSA and statistical matching in the same way as generalized linear models.

6. The Simulation Study

6.1. Data

All of the experiments were performed using three different populations. In addition, two different sampling strategies were selected for each one in order to recreate the behavior of the estimates under the lack of representativeness of the potentially sampled subpopulation and under selection mechanisms tied to individual features (e.g., voluntariness).

The first population, which will be referred to as P1, corresponds to the microdata of the Spanish Life Conditions Survey (2012 edition) [49]. It collects data about economic and life conditions variables for 28,610 adults living in Spain. We took the mean health, as reported by the individuals themselves on a scale from 1 to 5, as the objective variable to estimate. The algorithms were trained using the 56 most related variables, excluding "health issues in the last six months", "chronic conditions", "household income difficulties", and "civil status" (as they are too correlated with the target variable). The first sampling strategy for this population, which will be referred to as P1S1, was a simple random sampling excluding the individuals without internet access. In the second sampling strategy, P2S2, we also included a propensity to participate in the sample using the formula $Pr(yr) = \frac{yr^2 - 1900^2}{1996^2 - 1900^2}$, where yr is the year in which the individual was born. This way, linear models should have more problems learning the relations.

BigLucy [50], P2, was chosen as the second population. It consists of various financial variables of 85,396 industrial companies of a city for a particular fiscal year. The target variable chosen was the annual income in the previous fiscal year. The algorithms were trained using the size of the company (small, medium, or big), the number of employees, the company's income tax, and whether it is ISO certified. The first sampling strategy for this population, P2S1, was simple random sampling among the companies with SPAM options that are not small companies. This approach tested whether the models were able to correctly estimate the annual income for companies that were not in the training data. The second sampling strategy, P2S2, was simple random sampling among the companies with SPAM options, including a propensity to participate calculated as $Pr(taxes) = min(taxes^2/30, 1)$, where $taxes$ is the company's income tax. This scenario is similar but it implies a quadratic dependence.

The Bank Marketing Data Set [51], P3, is the third population. It includes information about 41,188 phone calls related to direct marketing campaigns of a Portuguese banking institution. Our goal is to predict the mean contact duration. A total of 18 variables were used for training. We excluded two of the dataset variables, the month, and whether the client has subscribed for a term deposit in order to make the inference more difficult. For the first sampling strategy, P3S1, we applied simple random sampling among the clients contacted more than three times. For the second sampling strategy, P3S2, we applied simple random sampling among the clients contacted more than twice. Surprisingly, filtering less led to worse estimations for some cases.

6.2. Simulation

Each population and sampling strategy was simulated using various sample sizes: 1000, 2000, and 5000. The same size is taken for the convenience sample and for the reference sample. For each sample size, 500 simulations were executed. In each simulation, PSA (using weights defined in Formula (4) in Section 3), PSA with stratification (using weights defined in Formula (10) in Section 3), and statistical matching estimates were obtained using several predictive algorithms.

For PSA (with and without stratification), the following classification algorithms were used: Logistic regression (*glm*), generalized linear model via penalized maximum likelihood (*glmnet*), Naive Bayes (*naivebayes*), k-Nearest Neighbors (*knn*), C4.5 decision tree (*J48*), Bagged Trees (*treebag*), Random Forests (*rf*), Gradient Boosting Machine (*gbm*), Model Averaged Neural Network (*avNNet*), Linear Discriminant Analysis (*lda*), Penalized Discriminant Analysis (*pda*), and Shrinkage Discriminant Analysis (*sda*).

For statistical matching, the following regression algorithms were used: linear regression (*glm*), Ridge regression with and without Bayesian priors (*bridge* and *ridge* respectively), LASSO regression via penalized maximum likelihood (*glmnet*), LARS-EN algorithm (*lasso*) and using Bayesian priors on the estimates (*blasso*), k-Nearest Neighbors (*knn*), Bagged Trees (*treebag*), Gradient Boosting Machine (*gbm*) and Bayesian-regularized Neural Networks (*brnn*).

These represent standard variants from different model types: Linear regression, penalized regression, Bayesian models, prototype models, trees, gradient boosting, neural networks, and discriminant analysis. All of the methods were trained using default hyperparameters, except for k-Nearest Neighbors, Naive Bayes, and C4.5, because their performance improved greatly after hypeparameter tuning. Said tuning was performed with bootstrap. The framework used for training, optimization, and prediction was *caret* [52], an R [53] package.

Different metrics are considered for evaluating each scenario: Relative mean bias, relative standard deviation, and relative Root Mean Square Error (RMSE).

$$RBias\ (\%) = \left(\frac{\sum_{i=1}^{500} \hat{p}_{yi}}{500} - p_y \right) \times \frac{100}{p_y} \tag{23}$$

$$RStandard\ deviation\ (\%) = \sqrt{\frac{\sum_{i=1}^{500} (\hat{p}_{yi} - \hat{\bar{p}}_y)^2}{499}} \times \frac{100}{p_y} \tag{24}$$

$$RMSE\ (\%) = \sqrt{RBias^2 + RSD^2}, \tag{25}$$

where p_y is the value of the target variable, $\hat{\bar{p}}_y$ is the mean of the 500 estimations of p_y, and $\hat{p}_{yi}$ is the estimation of p_y in the i-th simulation.

In order to rank each estimator, the mean efficiency, the median efficiency, and the number of times it has been among the best are measured. An estimator is considered to be among the best when its *RMSE* differs from the minimum *RMSE* by less than 1%. The efficiency is defined as follows:

$$Efficiency\ (\%) = \frac{Baseline - RMSE}{Baseline} \times 100, \tag{26}$$

where the baseline is the *RMSE* of using the sample average as the estimation.

To complete the comparison analysis, the results of relative bias, standard deviation, and RMSE were analyzed using linear mixed-effects regression. This approach provides estimates of the effect size of each adjustment method and algorithm. Datasets were considered random effects, while adjustment (matching, PSA, or PSA with propensity stratification) and algorithm (*glm*, *gbm*, *glmnet*, *knn*, and *treebag*, as they were the only algorithms used in all adjustments) were the fixed effects variables. All three response variables take non-negative values, and the interpretation is the same: The lower their value is, the better (less biased and/or less variable) the estimations are. Following this rule, negative Beta coefficients indicate that a given factor is contributing to better estimations, and vice versa for positive coefficients.

7. Results

Tables A1–A3 in the Appendix A show, respectively, the resulting biases, deviations, and RMSEs. In general, it can be observed that statistical matching outperforms PSA, which outperforms PSA with stratification. Nevertheless, all three methods consistently reduce the sample bias.

In terms of machine learning algorithms, basic linear models seem the most robust. Others, like Naive Bayes, can achieve outstanding results, but only for some cases. It is also interesting noting that C4.5 trees can even lead to worse estimations than simply using the sample average.

The final ranking confirming this impression can be seen in Table 1. Statistical matching with linear or ridge regression has the best mean efficiency and has been among the best more than the rest of approaches. This is not a surprise, since their simplicity avoids overfitting, presumably one of the main problems of matching. Ridge regression should be preferred if the data suffer from multicollinearity. Otherwise, linear regression alone is very effective (and faster).

Table 1. Mean and median efficiency (%) of each estimator and the number of times it has been among the best.

	Mean	Median	Best		Mean	Median	Best
matching ridge	61.5	63.8	10	psa gbm	30.7	28.8	3
matching glm	61.5	64.2	10	psa strat naive	30.5	32.1	3
matching glmnet	61	62.8	7	psa strat knn	25.6	24.7	0
matching brnn	57.3	61.7	7	matching lasso	24.6	14.4	3
matching blasso	57.1	59.9	6	psa strat glm	24.5	28.6	1
matching bridge	55.8	61.2	7	psa strat lda	23.4	27.5	0
matching knn	55.8	51.7	3	psa strat sda	23.2	27.2	0
psa glm	46.4	53.8	5	psa strat pda	23.2	27.2	0
psa sda	46.2	51.7	4	psa strat avNNet	21.8	23.4	0
psa glmnet	46.1	53.4	3	psa strat glmnet	21.7	28.1	1
psa lda	46	51.2	3	psa strat gbm	16.8	16.2	0
psa pda	45.7	51.7	4	psa treebag	10.1	11.6	0
psa naive	41.2	56.9	6	psa strat treebag	7.6	3.5	0
psa knn	38.5	42.4	3	psa strat rf	3.6	4.4	0
psa avNNet	34.2	33.2	0	psa rf	−4.5	7.8	0
matching gbm	32.2	34.9	0	psa strat J48	−23.9	3.8	0
matching treebag	31.4	49.1	1	psa J48	−36.7	7.9	0

The results of linear mixed-effects modeling can be consulted in Tables 2–4. It is noticeable how linear models, LASSO with LARS-EN algorithm, and k-Nearest Neighbors outperform Bagged Trees in all metrics (modulus of relative bias, relative standard deviation, and RMSE), while there is no evidence that GBM is different from any of them. Regarding adjustment methods, PSA (both with and without propensity stratification) showed significantly more bias and RMSE than statistical matching. In the case of standard deviation, there is also evidence that PSA without propensity stratification provides

higher values (deviation) than matching. Altogether, these results would indicate that matching has a larger effect on bias reduction than PSA.

Table 2. Linear mixed-effects model on the modulus of relative bias ($|RBias|$) considering adjustment methods and algorithms as fixed effects and datasets as random effects. Reference levels: Bagged Trees (*treebag*) algorithm and matching adjustment.

Coefficient	**Estimate**	**Std. Error**	**D. f.**	***t* Value**	**IC 95%**	***p*-Value**
(Intercept)	12.755	8.47	5.1276	1.506	[−8.857; 34.367]	0.19104
glm	−3.359	1.224	258.00	−2.745	[−5.769; −0.950]	0.00647
glmnet	−2.753	1.224	258.00	−2.250	[−5.163; −0.343]	0.02530
knn	−2.960	1.224	258.00	−2.419	[−5.370; −0.551]	0.01624
gbm	−0.624	1.224	258.00	−0.510	[−3.034; 1.785]	0.61042
psa	6.844	0.948	258.00	7.221	[4.978; 8.710]	5.79×10^{-12}
psa strat	9.601	0.948	258.00	10.129	[7.734; 11.467]	$<2 \times 10^{-16}$
Group	**Variance**	**Std. Dev.**				
Dataset	424.21	20.596				
Residual	40.42	6.358				
Dataset	**Sampling**	**Intercept**				
P1	P1S1	1.590				
P1	P1S2	4.522				
P2	P2S1	53.693				
P2	P2S2	13.029				
P3	P3S1	4.110				
P3	P3S2	−0.414				

Table 3. Linear mixed-effects model on relative standard deviation considering adjustment methods and algorithms as fixed effects and datasets as random effects. Reference levels: *treebag* algorithm and Matching adjustment.

Coefficient	**Estimate**	**Std. Error**	**D. f.**	***t* Value**	**IC 95%**	***p*-value**
(Intercept)	3.686	0.518	14.41	7.112	[2.578; 4.795]	4.5×10^{-6}
glm	−2.095	0.430	258.00	−4.866	[−2.942; −1.247]	2.0×10^{-6}
glmnet	−2.224	0.430	258.00	−5.167	[−3.071; −1.376]	4.8×10^{-7}
knn	−1.986	0.430	258.00	−4.614	[−2.833; −1.138]	6.2×10^{-6}
gbm	−2.020	0.430	258.00	−4.694	[−2.868; −1.173]	4.4×10^{-6}
psa	0.623	0.333	258.00	1.869	[−0.033; 1.280]	0.0628
psa strat	0.404	0.333	258.00	1.213	[−0.252; 1.061]	0.2262
Group	**Variance**	**Std. Dev.**				
Dataset	0.834	0.913				
Residual	5.002	2.236				
Dataset	**Sampling**	**Intercept**				
P1	P1S1	2.479				
P1	P1S2	2.703				
P2	P2S1	4.391				
P2	P2S2	4.064				
P3	P3S1	4.226				
P3	P3S2	4.254				

Table 4. Linear mixed-effects model on RMSE considering adjustment methods and algorithms as fixed effects and datasets as random effects. Reference levels: *treebag* algorithm and matching adjustment.

Coefficient	Estimate	Std. Error	D. f.	*t* value	IC 95%	*p*-value
(Intercept)	14.092	8.374	5.13	1.683	[−7.271; 35.455]	0.15174
glm	−4.094	1.222	258.00	−3.351	[−6.500; −1.688]	0.00093
glmnet	−3.515	1.222	258.00	−2.877	[−5.920; −1.109]	0.00435
knn	−3.639	1.222	258.00	−2.978	[−6.045; −1.233]	0.00317
gbm	−1.288	1.222	258.00	−1.054	[−3.694; 1.118]	0.29272
psa	6.744	0.946	258.00	7.126	[4.880; 8.607]	1.0×10^{-11}
psa strat	9.246	0.946	258.00	9.771	[7.383; 11.110]	$<2 \times 10^{-16}$
Group	**Variance**	**Std. Dev.**				
Dataset	414.5	20.359				
Residual	40.3	6.348				
Dataset	**Sampling**	**Intercept**				
P1	P1S1	2.436				
P1	P1S2	5.367				
P2	P2S1	54.549				
P2	P2S2	14.515				
P3	P3S1	5.949				
P3	P3S2	1.737				

8. Discussion

Nonprobability samples are increasingly common due to the growing internet penetration and the subsequent rise of online questionnaires. These questionnaries are a faster, less expensive, and more comfortable method of information collection in comparison to traditional ones. However, samples obtained with this technique deal with several sources of bias: Despite the increasing internet penetration, large population groups (less educated or elderly people) are still not properly represented. In addition, questionnaires are often administered with non-probabilistic sampling methods (e.g., snowballing), which imply that the selection is controlled by the interviewees themselves, causing a selection bias.

In this study, we focus on two of the proposed methods to reduce biases produced by nonprobability sampling: PSA and matching. We also compare the outcomes when the predictive modeling, required in both methods, is done through linear regression and through machine learning algorithms. PSA and matching require a probability sample on which the target variable has not been measured. The unit sampling performed in the simulations captures different self-selection scenarios in nonprobability sampling, while probability samples are drawn by simple random sampling with no sources of bias. This canonical representation is not usual, as reference samples are mostly drawn with complex sampling methods and the amount of bias is non-null. Further research could take into account these imperfect situations.

Results show that statistical matching provides better results than PSA on bias reduction and RMSE, regardless of the dataset and selection mechanism. In addition, linear models and k-nearest neighbors provided, on average, better results in terms of bias reduction than more complex models, such as GBM and Bagged Trees. These results are relevant since, even though there are comparative studies between adjustment techniques in nonprobability surveys [11,54], to the best of our knowledge, no comparison has been done before between these two methods.

Before closing, several limitations of our analysis should be mentioned. Given that the datasets used for simulation are real-life examples, we cannot ensure whether a selection bias mechanism is Missing At Random (MAR) or Missing Not At Random (MNAR), as the causality relationships are not known. It is known that the selection mechanism makes a difference in terms of how challenging bias reduction can be, but, in this study, it was not possible to assess.

In the near future, it is planned to explore how to combine PSA and statistical matching techniques. Shrinkage is a natural way to improve the available estimates in terms of the mean squared error that has been used by many authors in other contexts (e.g., [55–57]). The idea is to shrink the estimator $\hat{\theta}_{SM}$ towards the estimator $\hat{\theta}_{PSA}$ and obtain $\hat{\theta}_{srk} = K\hat{\theta}_{SM} + (1-K)\hat{\theta}_{PSA}$, where K is a constant satisfying $0 < K < 1$.

Another way can be considered by taking into account that most machine learning models allow weighting of the data used for training. Therefore, the weights obtained via PSA for the volunteer sample can be used when training the model used for statistical matching, since it is trained with said sample.

Author Contributions: Conceptualization, M.d.M.R.; formal analysis, M.d.M.R.; funding acquisition, M.d.M.R.; methodology, R.F.-G.; software, L.C.-M.; supervision, R.F.-G.; validation, L.C.-M.; writing, L.C.-M. All authors have read and agreed to the published version of the manuscript.

Funding: The work was supported by the Ministerio de Economia, Industria y Competitividad, Spain, under Grant MTM2015-63609-R and, in terms of the third author, an FPU grant from the Ministerio de Ciencia, Innovación y Universidades, Spain (FPU17/02177).

Acknowledgments: We would like to thank the anonymous referees for their remarks and comments that have improved the presentation and the contents of the paper.

Conflicts of Interest: The authors declare no conflict of interest.

Appendix A

Table A1. Relative mean bias (%) of each estimator for each population, sampling method, and sample size. The best values among the methods are shown in bold.

	P1S1			P1S2			P2S1			P2S2			P3S1			P3S2		
Estimator	**1000**	**2000**	**5000**	**1000**	**2000**	**5000**	**1000**	**2000**	**5000**	**1000**	**2000**	**5000**	**1000**	**2000**	**5000**	**1000**	**2000**	**5000**
baseline	8.4	8.5	8.5	12.9	12.8	12.8	70.6	70.4	70.4	32.7	32.6	32.9	13.4	13.2	13.3	5.9	5.8	5.9
matching blasso	3.3	3	2.7	6.6	5.8	5.2	24.6	**24.6**	**24.6**	12.6	12.6	12.6	8.9	5.1	2.7	3	**0.4**	2
matching bridge	3.4	3.1	2.8	6.9	6.1	5.3	24.5	**24.6**	**24.5**	12.3	12.6	12.7	9.1	4.8	2.3	4	**0.5**	1.5
matching brnn	**2.4**	2.3	2.2	**4.8**	4.5	4.3	25.3	**24.7**	**24.7**	13.3	13.3	13.3	4.3	2.5	**0**	**0.4**	2.1	4
matching gbm	5.3	5.3	5.4	8.8	8.9	8.9	46.2	46.7	47.1	17	17.4	17.5	**0.4**	2.3	5	5.8	5.3	3.2
matching glm	**2.6**	2.4	2.5	**4.7**	4.7	4.6	24.4	**24.6**	**24.7**	12.6	12.7	12.7	**0.6**	**0.5**	0.8	3.4	3.3	3
matching glmnet	**2.6**	2.6	2.5	**4.8**	4.7	4.8	25.5	25.6	25.6	12.7	12.9	12.8	**0.7**	0.8	0.9	3.2	3.2	3.2
matching knn	5	4.4	3.6	7.5	6.9	6.1	34.2	34.1	34	7	5.8	4.2	6.1	5.8	5.5	0.9	**0.5**	**0.3**
matching lasso	7.1	7.2	7.2	10.8	11	11.1	65.5	65.5	65.5	29.9	29.8	29.7	6.8	6.4	6.6	1.8	1.3	1.2
matching ridge	**2.4**	2.5	2.5	**4.7**	4.7	4.7	24.5	**24.6**	**24.7**	12.6	12.8	12.7	0.5	0.9	0.8	3.2	3.1	3.2
matching treebag	3.5	3.7	4.3	6.5	6.1	6.4	45.5	45.7	46	16.4	16.6	16.6	6.6	**0.1**	8.8	10.8	6	1.7
psa avNNet	3.9	4.2	3.9	6.3	6.7	6.6	66.8	64.2	62.6	8.6	6.9	10	10.5	8.9	10	5.1	4.5	4.6
psa gbm	5.9	5.7	5.4	9.4	9.2	9	66.1	66.6	67.2	15.1	15.3	15.6	10.9	11.1	11.5	1.3	1.3	1.4
psa glm	3.4	3.5	3.5	5.6	5.8	5.8	67.7	68	68.1	15.1	15.1	15.1	4.8	5.2	5.1	1.3	1.3	1.3
psa glmnet	3.7	3.6	3.6	5.9	6	5.9	66.6	66.7	66.9	15.2	15.1	15.1	5.4	5.6	5.4	1.2	1.1	1.1
psa J48	4.6	5	5.2	9.5	10.5	10.9	70.6	70.4	69.6	23.5	22.1	15.1	19.3	22.3	23.4	4.2	2.2	2
psa knn	4.3	4.2	4.1	7.3	7.1	7	68.4	68.5	66.1	18.8	18.2	13.2	7	7.5	7.9	**0.6**	**0.6**	**0.6**
psa lda	3.7	3.6	3.6	6.1	6.2	6.3	67.3	67.2	67.1	14.8	14.9	14.8	6.5	6.2	6.4	**0.2**	**0.3**	**0.2**
psa naive	**2.2**	**1**	**0.1**	**4.7**	**3.6**	**2.5**	**22.4**	**24.8**	29.8	**4.5**	6.4	7.1	3.4	4.1	4.3	4.3	4.4	5.3
psa pda	3.6	3.6	3.6	6.2	6.3	6.2	67.7	67.2	67	14.8	14.9	14.8	6.8	6.4	6.3	**0.5**	**0.2**	**0.3**
psa rf	7	7.1	7.4	11	11.1	11.4	117.2	125.3	134.9	26.4	30.7	30.9	11.6	11.9	12.4	4.1	4.8	5.1
psa sda	3.5	3.7	3.6	6.2	6.2	6.2	67.1	67	67	14.8	14.7	14.8	6.3	6.2	6.3	**0.5**	**0.3**	**0.3**
psa treebag	6.8	7	7.3	10.9	11	11.4	66.2	66	59.6	13.3	23.4	25.6	12.1	12.4	12.7	4.4	4.7	5.6
psa strat avNNet	6.4	6.5	6.5	9.4	9.5	9.4	68.1	68.5	67.6	17.9	16.4	18.3	12.8	13	12.8	3.5	3.2	3.5
psa strat gbm	7.2	7.1	7	11	10.7	10.6	66.3	65.2	64.1	22.8	22.9	23	12.2	12.2	12	3.9	4.2	4.2
psa strat glm	6.2	6.1	6.1	9	8.8	8.8	71.3	69.6	66.9	23	22.9	23	10.8	10.7	10.8	2.9	3	3
psa strat glmnet	6.3	6.2	6.1	9.1	8.9	8.9	77.3	78.7	81	23	22.9	23	10.9	10.9	10.9	3.1	3	3
psa strat J48	6.2	6.9	6.9	10.5	11.5	11.8	70.6	70.4	70	27.4	26.9	22.5	14.8	17.7	18.8	**0.5**	**0.2**	1.4
psa strat knn	6.7	6.6	6.6	9.1	9.1	9.1	69.5	69.7	67.6	16.9	15.5	11.2	11.4	11.1	11.1	3.4	3.6	3.5
psa strat lda	6.2	6.2	6.2	9.1	9.1	9	71.4	69.9	67.2	22.7	22.7	22.8	10.5	10.5	10.5	3.4	3.5	3.6
psa strat naive	5.6	5.5	5.1	6.4	5.8	5	79.8	79.2	77.4	**3.4**	**3.4**	**2.9**	10.4	10.4	10.4	2.5	1.9	**0.3**
psa strat pda	6.2	6.1	6.1	9.2	9.1	9.1	72	70	67.5	22.8	22.7	22.8	12.6	12.8	12.9	3.6	3.5	3.6
psa strat rf	7.4	7.5	8.1	12.5	12.4	12.1	82	83.3	85	21.4	26.8	30.2	10.5	10.6	10.5	5.5	5.5	5.6
psa strat sda	6.3	6.2	6.1	9.3	9.1	9.1	70.8	70	66.7	22.8	22.7	22.8	12.9	13.2	13.1	3.4	3.6	3.6
psa strat treebag	7.9	7.9	8.1	12.4	12.4	12.4	68.4	68.7	67.1	6.9	20.5	24.3	9.8	8.9	11.2	5.9	5.6	5.7

Table A2. Relative deviation (%) of each estimator for each population, sampling method, and sample size. The best values among the methods are in bold.

	P1S1			P1S2			P2S1			P2S2			P3S1			P3S2		
Estimator	**1000**	**2000**	**5000**	**1000**	**2000**	**5000**	**1000**	**2000**	**5000**	**1000**	**2000**	**5000**	**1000**	**2000**	**5000**	**1000**	**2000**	**5000**
baseline	1.1	0.8	0.5	1.3	0.9	0.5	1.7	1.2	0.7	2.4	1.7	1	3.1	1.9	0.8	3	2.1	1.2
matching blasso	1.5	1.1	0.6	1.8	1.3	0.8	1.5	1.2	0.7	1.7	1.2	0.7	3.9	3.1	1.5	3.8	2.7	1.6
matching bridge	1.6	1.1	0.7	2	1.4	1.1	1.7	2.3	0.7	1.8	1.2	0.7	5.1	3.5	1.6	4.1	3	1.6
matching brnn	1.6	1	0.6	1.9	1.4	0.8	2.1	1.1	0.7	1.9	1.2	0.8	6.2	5.1	2.3	5.3	4	1.6
matching gbm	1.4	1	0.6	1.7	1.2	0.7	1.3	0.9	0.6	1.7	1.2	0.7	7.6	4.3	1.6	6.6	5.1	3.6
matching glm	1.4	1	0.6	2	1.4	0.9	1.6	1.1	0.6	1.7	1.2	0.7	4.1	2.7	1.2	4	2.6	1.4
matching glmnet	1.5	1	0.6	2	1.3	0.8	1.6	1.1	0.6	1.6	1.2	0.7	3.9	2.8	1.1	4	2.5	1.4
matching knn	1.6	1	0.7	2	1.4	0.9	1.3	0.8	0.6	1.7	1.3	0.9	4.3	2.8	1.3	3.9	2.8	1.5
matching lasso	1.2	0.8	0.5	1.4	1	0.6	1.7	1.2	0.7	2	1.4	0.9	3.9	2.5	1.1	3.5	2.4	1.4
matching ridge	1.6	1	0.6	1.9	1.3	0.8	1.5	1.1	0.7	1.8	1.2	0.7	4	2.7	1.2	3.9	2.5	1.4
matching treebag	1.4	1.1	0.7	1.9	1.4	0.9	1.4	1	0.7	1.6	1.2	0.8	11	6.4	1.5	9	5.6	2
psa avNNet	1.6	1.1	0.9	2	1.4	1.1	8.8	14.6	16	6.8	3.2	5.3	5	6	3.9	3.4	3	2
psa gbm	1.3	0.9	0.5	1.6	1	0.7	2.7	1.8	1.1	1.8	1.3	0.7	3.9	2.6	1.1	4.1	2.7	1.4
psa glm	1.5	1	0.6	2	1.4	0.8	3	2	1.3	1.7	1.2	0.8	4.1	2.7	1.2	4	2.6	1.3
psa glmnet	1.5	1	0.6	2	1.3	0.8	2.7	1.9	1.3	1.7	1.2	0.8	4	2.6	1.1	3.7	2.6	1.4
psa J48	2.6	1.8	1.2	2.9	2.1	1.2	1.7	1.2	1.6	8.7	8.5	6.5	21.9	19.9	18.6	25.2	17.9	11.5
psa knn	1.9	1.2	0.7	2.1	1.6	0.9	3.4	2.6	1.4	3.4	2.5	1.6	4.9	3.3	1.5	4.5	3.2	1.9
psa lda	1.5	1	0.6	1.9	1.2	0.8	3.6	2.6	1.6	1.9	1.2	0.7	3.5	2.4	1	3.6	2.4	1.3
psa naive	4.3	2.3	1.2	3.4	2.4	1.4	8.4	8.4	5.6	17.2	4.8	0.8	13.1	8	3.8	10.7	6.4	5.9
psa pda	1.5	1	0.5	1.8	1.2	0.7	3.9	2.8	1.6	1.7	1.2	0.7	3.6	2.3	1	3.6	2.6	1.3
psa rf	1.3	1	0.6	1.5	1.1	0.7	11.7	9.7	8.7	5.2	3.6	3.2	5.2	3.6	2.4	4.5	3.3	2.1
psa sda	1.4	1	0.6	1.8	1.2	0.7	3.9	2.7	1.6	1.7	1.2	0.7	3.5	2.4	1	3.6	2.5	1.3
psa treebag	1.6	1.2	1.3	1.8	1.2	0.8	9.7	13.9	8.8	21.1	8.7	4.2	5.7	4.3	2.8	5.3	4	2.4
psa strat avNNet	1.2	0.8	0.5	1.6	1.1	0.7	5.4	5.5	7.2	3.6	1.9	3.7	3.2	2.3	1.1	4	2.7	1.6
psa strat gbm	1.2	0.8	0.5	1.4	1	0.6	4.9	4.4	3.6	1.5	1.1	0.8	3.5	2.2	0.9	3.5	2.2	1.2
psa strat glm	1.2	0.8	0.5	1.5	1	0.6	7.8	6.7	4.5	1.6	1.1	0.8	3.2	2.1	0.9	3.5	2.1	1.2
psa strat glmnet	1.2	0.8	0.5	1.5	1.1	0.6	6.4	4.9	2.5	1.7	1.1	0.7	3.2	1.9	0.9	3.3	2.2	1.2
psa strat J48	2.2	1.1	0.7	2.4	1.5	0.8	2.1	1.4	4.2	5.4	4.9	4	15.4	12.3	10.5	20.1	14.6	10
psa strat knn	1.2	0.9	0.5	1.7	1.2	0.7	2.7	1.9	1.4	5	3.7	2	3.2	2.3	1.2	3.4	2.5	1.3
psa strat lda	1.2	0.8	0.5	1.5	1	0.6	7.1	6.6	4.8	1.7	1.2	0.8	3.2	2	0.9	3.3	2.3	1.3
psa strat naive	1.6	1.1	0.6	2.7	1.7	0.9	4.1	3.6	2.7	2.8	2	1.3	3.1	2	0.9	6	4.8	4
psa strat pda	1.2	0.8	0.5	1.4	1	0.6	7.6	6.8	4.9	1.7	1.3	0.8	2.9	1.9	0.9	3.2	2.2	1.2
psa strat rf	1.2	0.8	0.4	1.4	0.9	0.6	3.2	2.5	1.7	4.2	3.3	2.9	3.1	1.9	0.9	3.3	2.1	1.2
psa strat sda	1.2	0.8	0.5	1.5	1.1	0.6	7.5	7.1	4.4	1.8	1.3	0.8	3	2	1	3.3	2.2	1.2
psa strat treebag	1.2	0.8	0.5	1.4	0.9	0.6	5.6	7.7	4.5	20.8	8.8	4	10.5	10.4	3.7	3.1	2.2	1.3

Table A3. Relative Root Mean Square Error (RMSE) (%) of each estimator for each population, sampling method, and sample size. The best values among the methods are shown in bold.

	P1S1			P1S2			P2S1			P2S2			P3S1			P3S2		
Estimator	**1000**	**2000**	**5000**	**1000**	**2000**	**5000**	**1000**	**2000**	**5000**	**1000**	**2000**	**5000**	**1000**	**2000**	**5000**	**1000**	**2000**	**5000**
baseline	8.5	8.5	8.5	12.9	12.9	12.8	70.6	70.4	70.5	32.8	32.7	32.9	13.8	13.4	13.3	6.6	6.2	6
matching blasso	3.6	3.2	2.8	6.9	6	5.3	24.7	**24.6**	**24.6**	12.7	12.7	12.6	9.7	6	3	4.8	**2.7**	2.6
matching bridge	3.8	3.3	2.8	7.1	6.3	5.4	24.6	**24.7**	**24.5**	12.5	12.6	12.7	10.4	5.9	2.8	5.7	3	2.2
matching brnn	**2.9**	**2.5**	2.5	**5.2**	**4.7**	4.4	25.4	**24.7**	**24.7**	13.4	13.4	13.3	7.6	5.6	2.3	5.3	4.5	4.3
matching gbm	5.5	5.4	5.4	9	9	8.9	46.2	46.7	47.1	17.1	17.4	17.5	7.6	4.9	5.2	8.8	7.3	4.8
matching glm	**3**	**2.6**	2.5	**5.1**	**4.9**	4.7	**24.4**	**24.6**	**24.7**	12.7	12.7	12.7	**4.2**	**2.8**	**1.4**	5.2	4.2	3.3
matching glmnet	**3**	**2.8**	2.6	**5.2**	**4.9**	4.9	25.5	25.6	25.6	12.8	12.9	12.8	**4**	**2.9**	**1.4**	5.2	4	3.4
matching knn	5.3	4.6	3.6	7.8	7	6.2	34.2	34.1	34	7.2	6	4.3	7.5	6.5	5.7	**4**	**2.9**	**1.5**
matching lasso	7.2	7.3	7.3	10.9	11.1	11.1	65.6	65.5	65.5	30	29.8	29.7	7.8	6.8	6.7	**3.9**	**2.7**	**1.8**
matching ridge	**2.9**	**2.7**	2.6	**5.1**	**4.9**	4.8	**24.5**	**24.6**	**24.7**	12.7	12.9	12.7	**4**	**2.9**	**1.5**	5.1	4	3.5
matching treebag	3.8	3.8	4.3	6.7	6.2	6.4	45.5	45.7	46	16.5	16.6	16.7	12.8	6.4	9	14	8.2	2.7
psa avNNet	4.2	4.3	4	6.6	6.8	6.7	67.4	65.8	64.6	11	7.6	11.3	11.7	10.8	10.7	6.1	5.4	5
psa gbm	6.1	5.8	5.5	9.5	9.3	9	66.1	66.6	67.2	15.2	15.3	15.6	11.6	11.4	11.5	4.3	3	2
psa glm	3.7	3.6	3.5	6	6	5.9	67.8	68.1	68.1	15.2	15.1	15.2	6.4	5.8	5.2	4.2	3	1.9
psa glmnet	4	3.8	3.7	6.2	6.1	6	66.7	66.8	66.9	15.3	15.1	15.1	6.7	6.2	5.6	**3.9**	**2.8**	**1.8**
psa J48	5.3	5.3	5.4	9.9	10.7	10.9	70.6	70.4	69.6	25	23.7	16.4	29.2	29.9	29.9	25.5	18.1	11.7
psa knn	4.7	4.4	4.2	7.6	7.3	7.1	68.5	68.5	66.1	19.1	18.4	13.3	8.5	8.2	8	4.5	3.3	2
psa lda	4	3.7	3.6	6.4	6.3	6.3	67.4	67.2	67.2	14.9	14.9	14.8	7.4	6.7	6.4	**3.6**	**2.4**	**1.3**
psa naive	4.8	**2.6**	**1.2**	5.8	4.3	**2.9**	**23.9**	26.2	30.3	17.8	8	7.1	13.5	9	5.7	11.6	7.8	8
psa pda	3.9	3.7	3.6	6.5	6.5	6.3	67.8	67.2	67	14.9	15	14.9	7.7	6.8	6.3	**3.7**	**2.6**	**1.3**
psa rf	7.1	7.1	7.4	11.1	11.2	11.4	117.8	125.7	135.2	26.9	30.9	31.1	12.7	12.4	12.7	6.1	5.8	5.5
psa sda	3.8	3.8	3.7	6.5	6.3	6.3	67.2	67.1	67	14.9	14.8	14.8	7.3	6.6	6.4	**3.6**	**2.5**	**1.3**
psa treebag	7	7.1	7.4	11.1	11.1	11.4	66.9	67.4	60.2	25	24.9	26	13.3	13.1	13	6.9	6.1	6.1
psa strat avNNet	6.5	6.6	6.5	9.5	9.6	9.4	68.3	68.7	68	18.3	16.5	18.6	13.2	13.2	12.8	5.3	4.2	3.8
psa strat gbm	7.3	7.2	7	11.1	10.8	10.7	66.5	65.3	64.2	22.9	22.9	23	12.7	12.4	12	5.2	4.8	4.4
psa strat glm	6.3	6.2	6.2	9.1	8.8	8.8	71.7	69.9	67.1	23.1	22.9	23	11.2	10.9	10.8	4.5	3.7	3.2
psa strat glmnet	6.4	6.2	6.2	9.2	9	9	77.5	78.8	81	23.1	22.9	23	11.3	11.1	10.9	4.5	3.8	3.3
psa strat J48	6.6	7	7	10.7	11.6	11.9	70.6	70.5	70.2	27.9	27.3	22.9	21.4	21.5	21.6	20.1	14.6	10.1
psa strat knn	6.8	6.7	6.6	9.3	9.2	9.1	69.5	69.7	67.6	17.6	15.9	11.3	11.8	11.3	11.2	4.8	4.3	3.7
psa strat lda	6.3	6.2	6.2	9.2	9.1	9	71.7	70.2	67.4	22.7	22.7	22.9	11	10.7	10.5	4.8	4.2	3.8
psa strat naive	5.9	5.6	5.2	6.9	6	5.1	79.9	79.3	77.4	**4.4**	**3.9**	**3.2**	10.8	10.6	10.5	6.5	5.2	4
psa strat pda	6.3	6.2	6.2	9.3	9.2	9.1	72.4	70.4	67.7	22.8	22.7	22.9	12.9	12.9	13	4.9	4.2	3.8
psa strat rf	7.5	7.6	8.1	12.5	12.5	12.1	82	83.3	85	21.8	27	30.3	10.9	10.8	10.5	6.4	5.9	5.8
psa strat sda	6.4	6.2	6.2	9.4	9.1	9.1	71.2	70.3	66.8	22.8	22.8	22.8	13.2	13.3	13.1	4.7	4.3	3.8
psa strat treebag	8	8	8.1	12.5	12.4	12.4	68.6	69.1	67.2	21.9	22.3	24.6	14.4	13.7	11.8	6.6	6	5.9

References

1. Rada, D. Ventajas e inconvenientes de la encuesta por Internet. *Pap. Rev. Sociol.* **2012**, *97*, 193–223.
2. Elliott, M.R.; Valliant, R. Inference for nonprobability samples. *Stat. Sci.* **2017**, *32*, 249–264. [CrossRef]
3. Meng, X.L. Statistical paradises and paradoxes in big data (I), Law of large populations, big data paradox, and the 2016 US presidential election. *Ann. Appl. Stat.* **2018**, *12*, 685–726. [CrossRef]
4. Royall, R.M.; Herson, J. Robust estimation in finite populations I. *J. Am. Stat. Assoc.* **1973**, *68*, 880–889. [CrossRef]
5. Valliant, R.; Dorfman, A.H.; Royall, R.M. *Finite Population Sampling and Inference: A Prediction Approach*; John Wiley: New York, NY, USA, 2000; No. 04, QA276. 6, V3.
6. Buelens, B.; Burger, J.; van den Brakel, J.A. Comparing inference methods for non-probability samples. *Int. Stat. Rev.* **2018**, *86*, 322–343. [CrossRef]
7. Lee, S. Propensity score adjustment as a weighting scheme for volunteer panel web surveys. *J. Off. Stat.* **2006**, *22*, 329–349.
8. Lee, S.; Valliant, R. Estimation for volunteer panel web surveys using propensity score adjustment and calibration adjustment. *Sociol. Methods Res.* **2009**, *37*, 319–343. [CrossRef]
9. Valliant, R.; Dever, J.A. Estimating propensity adjustments for volunteer web surveys. *Sociol. Methods Res.* **2011**, *40*, 105–137. [CrossRef]
10. Ferri-García, R.; Rueda, M.D.M. Efficiency of propensity score adjustment and calibration on the estimation from non-probabilistic online surveys. *Stat. Oper. Res. Trans.* **2018** , *1*, 159–182.
11. Valliant, R. Comparing Alternatives for Estimation from Nonprobability Samples. *J. Surv. Stat. Methodol.* **2020**, *8*, 231–263. [CrossRef]
12. Ferri-García, R.; Rueda, M.D.M. Propensity score adjustment using machine learning classification algorithms to control selection bias in online surveys. *PLoS ONE* **2020**, *15*, e0231500. [CrossRef] [PubMed]
13. Couper, M.P. The future of modes of data collection. *Public Opin. Q.* **2011**, *75*, 889–908. [CrossRef]
14. Rosenbaum, P.R.; Rubin, D.B. The central role of the propensity score in observational studies for causal effects. *Biometrika* **1983** , *70*, 41–55. [CrossRef]
15. Taylor, H. Does internet research work? *Int. J. Mark.* **2000**, *42*, 1–11. [CrossRef]
16. Taylor, H.; Bremer, J.; Overmeyer, C.; Siegel, J.W.; Terhanian, G. The record of internet-based opinion polls in predicting the results of 72 races in the November 2000 US elections. *Int. J. Mark. Res.* **2001**, *43*, 127–135.
17. Schonlau, M.; Van Soest, A.; Kapteyn, A. Are 'Webographic' or attitudinal questions useful for adjusting estimates from Web surveys using propensity scoring? *Surv. Res. Methods* **2007**, *1*, 155–163. [CrossRef]
18. Schonlau, M.; Couper, M.P. Options for conducting web surveys. *Stat. Sci.* **2017**, *32*, 279–292. [CrossRef]
19. Lee, B.K.; Lessler, J.; Stuart, E.A. Improving propensity score weighting using machine learning. *Stat. Med.* **2010**, *29*, 337–346. [CrossRef]
20. Phipps, P.; Toth, D. Analyzing establishment nonresponse using an interpretable regression tree model with linked administrative data. *Ann. Appl. Stat.* **2012**, *6*, 772–794. [CrossRef]
21. Buskirk, T.D.; Kolenikov, S. Finding respondents in the forest: A comparison of logistic regression and random forest models for response propensity weighting and stratification. *Surv. Methods Insights Field* **2015**, 1–17. [CrossRef]
22. Rivers, D. Sampling for Web Surveys. In Proceeding of the Joint Statistical Meetings, Salt Lake City, UT, USA, 1 August 2007.
23. Beaumont, J.F.; Bissonnette, J. Variance Estimation under Composite Imputation. The methodology behind SEVANI. *Surv. Methodol.* **2011**, *37*, 171–179.
24. Fuller, W.A. Regression estimation for survey samples. *Surv. Methodol.* **2002**, *28*, 5–24.
25. Hoerl, A.E.; Kennard, R.W. Ridge regression: Biased estimation for nonorthogonal problems. *Technometrics* **1970**, 12, 55–67. [CrossRef]
26. Hsiang, T. A Bayesian view on ridge regression. *J. R. Soc. Ser. D* **1975**, *24*, 267–268. [CrossRef]
27. Tibshirani, R. Regression shrinkage and selection via the lasso. *J. R. Stat. Soc. Ser. B* **1996**, *58*, 267–288. [CrossRef]

28. Park, T.; Casella, G. The bayesian lasso. *J. Am. Stat. Assoc.* **2008**, *103*, 681–686. [CrossRef]
29. Van Houwelingen, J. Shrinkage and penalized likelihood as methods to improve predictive accuracy. *Stat. Neerl.* **2001**, *55*, 17–34. [CrossRef]
30. James, G.; Witten, D.; Hastie, T.; Tibshirani, R. *An Introduction to Statistical Learning*; Springer: Berlin, Germany, 2013.
31. Hastie, T.; Buja, A.; Tibshirani, R. Penalized discriminant analysis. *Ann. Stat.* **1995**, *23*, 73–102. [CrossRef]
32. Hastie, T.; Tibshirani, R.; Friedman, J. *The Elements of Statistical Learning: Data Mining, Inference, and Prediction*; Springer Science & Business Media: Berlin, Germany, 2009.
33. Breiman, L. Bagging predictors. *Mach. Learn.* **1996**, *24*, 123–140. [CrossRef]
34. Sutton, C.D. Classification and regression trees, bagging, and boosting. *Handb. Stat.* **2005**, *24*, 303–329.
35. Breiman, L. Random forests. *Mach. Learn.* **2001**, *45*, 5–32. [CrossRef]
36. Niculescu-Mizil, A.; Caruana, R. Predicting good probabilities with supervised learning. In Proceedings of the 22nd International Conference on Machine Learning, Bergamo, Italy, 5–7 September 2015; pp. 625–632.
37. Breiman, L. *Arcing the Edge. Tech. Rep., Technical Report 486*; Statistics Department, University of California: Berkeley, CA, USA, 1997.
38. Friedman, J.H. Greedy function approximation: A gradient boosting machine. *Ann. Stat.* **2001**, 1189–1232. [CrossRef]
39. Touzani, S.; Granderson, J.; Fernandes, S. Gradient boosting machine for modeling the energy consumption of commercial buildings. *Energy Build.* **2018**, *158*, 1533–1543. [CrossRef]
40. Peterson, L.E. K-nearest neighbor. *Scholarpedia* **2009**, *4*, 1883. [CrossRef]
41. García, S.; Luengo, J.; Herrera, F. *Data Preprocessing in Data Mining*; Springer International Publishing: Cham, Switzerland, 2015.
42. Okut, H. Bayesian regularized neural networks for small n big p data. In *Artificial Neural Networks-Models and Applications*; IN-TECH: Munich, Germany, 2016.
43. Rumelhart, D.E.; Hinton, G.E.; Williams, R.J. Learning representations by back-propagating errors. *Nature* **1986**, *323*, 533–536. [CrossRef]
44. Ripley, B.D. *Pattern Recognition and Neural Networks*; Cambridge University Press: Cambridge, UK, 1996.
45. Breidt, F.J.; Opsomer, J.D. Model-assisted survey estimation with modern prediction techniques. *Stat. Sci.* **2017**, *32*, 190–205. [CrossRef]
46. LeCun, Y.; Bengio, Y.; Hinton, G. Deep learning. *Nature* **2015**, *521*, 436–444. [CrossRef]
47. Baesens, B.; Viaene, S.; Poel, V.d.D.; Vanthienen, J.; Dedene, G. Bayesian neural network learning for repeat purchase modelling in direct marketing. *Eur. J. Oper. Res.* **2002**, *138*, 191–211. [CrossRef]
48. Csáji, B.C. Approximation with artificial neural networks. *Fac. Sci. Etvs Lornd Univ. Hung.* **2001**, *24*, 7.
49. National Institute of Statistics (INE). Life Conditions Survey. Microdata. Available online: https://www.ine.es/dyngs/INEbase/en/operacion.htm?c=Estadistica_C&cid=1254736176807menu=resultados&&idp=1254735976608#!tabs-1254736195153 (accessed on 30 May 2020).
50. Gutiérrez, H.A. *Estrategias de Muestreo Diseño de Encuestas y Estimacion de Parametros*; Universidad Santo Tomas: Bogota, Colombia, 2009.
51. Moro, S.; Cortez, P.; Rita, P. A data-driven approach to predict the success of bank telemarketing. *Decis. Support Syst.* **2014**, *62*, 22–31. [CrossRef]
52. Kuhn, M. *Caret: Classification and Regression Training*; R Package Version 6.0-81; R Foundation for Statistical Computing: Vienna, Austria, 2018.
53. R Core Team. *R: A Language and Environment for Statistical Computing*; R Foundation for Statistical Computing: Vienna, Austria, 2018.
54. Chen, J.K.T.; Valliant, R.L.; Elliott, M.R. Calibrating non-probability surveys to estimated control totals using LASSO, with an application to political polling. *J. R. Stat. Soc. Ser. C (Appl. Stat.)* **2019**, *68*, 657–681. [CrossRef]
55. James, W.; Stein, C. Estimation with quadratic loss. In *Proceedings of the Fourth Berkeley Symposium on Mathematical Statistics and Probability*; University of California Press: Berkeley, CA, USA, 1961; Volume 1, pp. 311–319.

56. Copas, J.B. The shrinkage of point scoring methods. *J. R. Stat. Soc. Ser. C (Appl. Stat.)* **1993**, *42*, 315–331. [CrossRef]
57. Arcos, A.; Contreras, J.M.; Rueda, M. A Novel Calibration Estimator in Social Surveys. *Sociol. Methods Res.* **2014** *43*, 465–489. [CrossRef]