A Debiased Machine Learning Single-Imputation Framework for Item Nonresponse in Surveys

David Haziza

Department of Mathematics and Statistics
University of Ottawa

Joint work with
Mehdi Dagdoug
(McGill University)

4th Workshop on Methodologies for Official Statistics
Rome, Italy

December 1, 2025

The setup

- $U = \{1, 2, ..., N\}$: finite population of size N.
- Y: Survey variable
- Goal: estimate the finite population mean

$$\mu = \frac{1}{N} \sum_{k \in U} y_k.$$

- S: sample of size n selected according to a probability sampling design with first-order inclusion probabilities $\pi_k = \mathbb{P}(I_k = 1) > 0$, where $I_k = 1$ if $k \in S$ and $I_k = 0$, otherwise.
- $\mathbf{I} = (I_1, \cdots, I_k, \cdots, I_N)^\top$: vector of sample selection indicators
- Full sample (Narain–Horvitz–Thompson) estimator of μ:

$$\widehat{\mu}_{\mathrm{HT}} = \frac{1}{N} \sum_{k \in S} w_k y_k \quad \text{with } w_k = 1/\pi_k$$

- Design-unbiased: $\mathbb{E}_p(\widehat{\mu}_{\mathrm{HT}}) = \mu$ for any survey variable Y.

Item nonresponse: Imputation

- The survey variable Y is prone to missing values.
- Let r_k be the response indicator such that

$$r_k = \begin{cases} 1 & \text{if } y_k \text{ is observed,} \\ 0 & \text{if } y_k \text{ is missing.} \end{cases}$$

- $\mathbf{r} = (r_1, \cdots, r_k, \cdots, r_N)^\top$: vector of response indicators.
- Set of respondents: $S_r = \{k \in S; r_k = 1\}$.
- Set of nonrespondents: $S_m = \{k \in S; r_k = 0\}$.

Assumptions

- $\mathbf{x}_k$: vector of fully observed variables associated with $k \in S$.
- Data are assumed to be MAR (Rubin, 1976):

$$\mathbb{P}\left[r_k = 1 | y_k, \mathbf{x}_k\right] = \mathbb{P}\left[r_k = 1 | \mathbf{x}_k\right] \equiv p(\mathbf{x}_k).$$

- Positivity assumption: $\mathbb{P}\left[r_k = 1 | \mathbf{x}_k\right] > 0$.
- Imputation model:

$$y_k = m(\mathbf{x}_k) + \epsilon_k, \qquad k \in U$$

  - $m(\mathbf{x})$: unknown and assumed to be continuous.
  - $\{\epsilon_k\}_{k \in U}$: independent such that $\mathbb{E}\left[\epsilon_k | \mathbf{x}_k\right] = 0$ and $\mathbb{V}\left[\epsilon_k | \mathbf{x}_k\right] = \sigma_k^2$.

Imputed estimator

- Imputed values:

$$y_k^* := \widehat{m}(\mathbf{x}_k), \qquad k \in S_m,$$

with $\widehat{m}$ an estimator of m fitted on

$$D_r := \{(\mathbf{x}_k, y_k)\,; k \in S_r\}\,.$$

- Imputed estimator of μ:

$$\widehat{\mu}_{imp} := \frac{1}{N}\left\{\sum_{k\in S} w_k r_k y_k + \sum_{k\in S} w_k (1-r_k)\widehat{m}(\mathbf{x}_k)\right\}.$$

- $\widehat{m}(\mathbf{x}_k)$: nuisance function, i.e., not of primary interest
- What are the properties of $\widehat{\mu}_{imp}$ and how to draw statistical inferences?

Parametric imputation

$$\widehat{\mu}_{imp} := \frac{1}{N}\left\{\sum_{k\in S} w_k r_k y_k + \sum_{k\in S} w_k (1-r_k)\widehat{m}(\mathbf{x}_k)\right\}$$

- For instance, linear regression imputation:

$$\widehat{m}(\mathbf{x}_k) = \mathbf{x}_k^\top \widehat{\boldsymbol{\beta}}$$

- First moment of the imputation model is correctly specified:
  - Correct functional form;
  - Correct specification of the $\mathbf{x}$-vector (interactions, terms accounting for curvature, etc.);
  - $\widehat{\mu}_{\text{imp}}$ is $\sqrt{n}$–consistent and asymptotically normal;
  - Several variance estimation methods have been developed (e.g., Haziza and Vallée, 2020).
- Misspecification of the first moment of the imputation model :
  - Bias does not vanish, even asymptotically $\longrightarrow$ Invalid inferences!

ML-Based Imputation

$$\widehat{\mu}_{imp} := \frac{1}{N}\left\{\sum_{k\in S} w_k r_k y_k + \sum_{k\in S} w_k(1-r_k)\widehat{m}(\mathbf{x}_k)\right\}$$

Nice features:

- Offers robustness against model misspecification such as incorrect functional form, omitted interaction terms, and unmodeled nonlinearities

Inference is challenging when using ML to obtain $\widehat{m}(\mathbf{x}_k)$:

- ML estimators typically converge slower than $n^{-1/2}$ $\longrightarrow$ $\widehat{\mu}_{imp}$ is consistent ;
- A CLT usually fails to hold $\longrightarrow$ difficult to justify confidence intervals;
- Lack of Neyman orthogonality $\longrightarrow$ $\widehat{\mu}_{\text{imp}}$ is affected by first-order errors in $\widehat{m}(\mathbf{x}_k)$
  $\longrightarrow$ Estimating the variance of $\widehat{\mu}_{\text{imp}}$ is very challenging for arbitrary ML imputation methods

Double-debiased imputation based on machine learning

- Double debiased ML framework: Chernozhukov et al. (2018)
- Relies on two key ingredients:
  - Neyman orthogonality
    - Small errors in nuisance parts have only second-order (negligible) impact on the target parameter (here, the population mean)
    - It provides "local immunity".
  - Crossfitting
- Doubly robust estimators: satisfy the Neyman orthogonality property.
- Originally, DR estimators were proposed to provide some robustness against the misspecification of either the imputation model or the nonresponse model: e.g., Robins et al. (1994), Kott (1994), Haziza and Rao (2006), Cao et al. (2009), and Kim and Haziza (2014).

Oracle doubly robust estimator

- We start by considering an oracle doubly robust (DR) estimator of μ:

$$\begin{aligned}\widehat{\mu}_{dr}(m,p) &= \frac{1}{N}\left\{\sum_{k\in S} w_k m(\mathbf{x}_k) + \sum_{k\in S} w_k \frac{r_k}{p(\mathbf{x}_k)}\{y_k - m(\mathbf{x}_k)\}\right\}, \\ &= \frac{1}{N}\sum_{k\in S} w_k \eta_k,\end{aligned}$$

where η_k is a pseudo-value associated with unit k:

$$\eta_k = m(\mathbf{x}_k) + \frac{r_k}{p(\mathbf{x}_k)}\{y_k - m(\mathbf{x}_k)\} \quad \text{for } k \in S.$$

Oracle doubly robust estimator: Its properties

- Unbiasedness:

$$\mathbb{E}\left\{\widehat{\mu}_{dr}(m,p)-\mu\right\}=0$$

$\mathbb{E}(.)$: expectation with respect to the joint distribution induced by the outcome regression model, the sampling design, and the propensity score model.

- Root-n consistency:

$$\widehat{\mu}_{dr}(m,p)-\mu=O_p\left(n^{-1/2}\right)$$

Oracle doubly robust estimator: Its properties

- Variance based on the reverse approach (Shao and Steel, 1999):

$$\mathbb{V}_{tot}\left(\widehat{\mu}\right) = \mathbb{E}\mathbb{V}\left(\widehat{\mu} \mid \mathbf{X}, \mathbf{r}, \mathbf{y}\right) + \mathbb{E}\mathbb{V}\left\{\mathbb{E}(\widehat{\mu} - \mu \mid \mathbf{X}, \mathbf{r}, \mathbf{y}) \mid \mathbf{X}, \mathbf{r}\right\},$$

  where $\mathbf{y} = (y_1, \ldots, y_N)^\top$ and $\mathbf{X}$ is the matrix whose kth row is $\mathbf{x}_k^\top$.

- Total variance of $\widehat{\mu}_{dr}(m, p)$ (assuming n/N negligible):

$$\mathbb{V}_{tot}\left\{\widehat{\mu}_{dr}(m, p)\right\} = \frac{1}{N^2} \sum_{k \in U} \sum_{\ell \in U} (\pi_{k\ell} - \pi_k \pi_\ell) \frac{\eta_k}{\pi_k} \frac{\eta_\ell}{\pi_\ell}$$

- Special case: SRSWOR

$$\mathbb{V}_{tot}\left\{\widehat{\mu}_{dr}(m, p)\right\} = \left(1 - \frac{n}{N}\right) \frac{S_\eta^2}{n}$$

Oracle doubly robust estimator: Its properties

- Unbiased estimator of $\mathbb{V}_{tot}\{\widehat{\mu}_{dr}(m,p)\}$:

$$\widehat{\mathbb{V}}_{tot}\{\widehat{\mu}_{dr}(m,p)\} := \frac{1}{N^2}\sum_{k\in S}\sum_{\ell\in S}\frac{(\pi_{k\ell}-\pi_k\pi_\ell)}{\pi_{k\ell}}\frac{\eta_k}{\pi_k}\frac{\eta_\ell}{\pi_\ell},$$

where

$$\eta_k = m(\mathbf{x}_k) + \frac{r_k}{p(\mathbf{x}_k)}\{y_k - m(\mathbf{x}_k)\} \quad \text{for } k \in S.$$

- Special case: SRSWOR

$$\widehat{\mathbb{V}}_{tot}\{\widehat{\mu}_{dr}(m,p)\} = \left(1-\frac{n}{N}\right)\frac{s_\eta^2}{n}$$

- Under some regularity conditions:

$$\frac{\widehat{\mu}_{dr}(m,p)-\mu}{\sqrt{\widehat{\mathbb{V}}_{tot}\{\widehat{\mu}_{dr}(m,p)\}}} \to \mathcal{N}(0,1).$$

DR estimation based on estimated nuisance functions

- Consider a DR estimator:

$$\begin{aligned}\widehat{\mu}_{dr}(\widehat{m},\widehat{p}) &= \frac{1}{N}\left\{\sum_{k\in S} w_k \widehat{m}(\mathbf{x}_k) + \sum_{k\in S} w_k \frac{r_k}{\widehat{p}(\mathbf{x}_k)}\{y_k - \widehat{m}(\mathbf{x}_k)\}\right\}, \\ &= \frac{1}{N}\sum_{k\in S} w_k \widehat{\eta}_k,\end{aligned}$$

where

$$\widehat{\eta}_k = \widehat{m}(\mathbf{x}_k) + \frac{r_k}{\widehat{p}(\mathbf{x}_k)}\{y_k - \widehat{m}(\mathbf{x}_k)\} \quad \text{for } k \in S.$$

- $\widehat{m}(\mathbf{x}_k)$: predicted value for the missing $y_k, k \in S$, based on some ML method
- $\widehat{p}(\mathbf{x}_k)$: estimated response probability for $k \in S_r$ based on some ML method.
- Neither $\widehat{m}(\mathbf{x})$ nor $\widehat{p}(\mathbf{x})$ is of primary interest $\longrightarrow$ can be viewed as nuisance functions

DR estimation based on estimated nuisance functions

- Assuming n/N is negligible, an estimator of the variance of $\widehat{\mu}_{dr}(\widehat{m}, \widehat{p})$ is

$$\widehat{\mathbb{V}}_{tot}\left\{\widehat{\mu}_{dr}(\widehat{m}, \widehat{p})\right\} := \frac{1}{N^2} \sum_{k \in S} \sum_{\ell \in S} \frac{(\pi_{k\ell} - \pi_k \pi_\ell)}{\pi_{k\ell}} \frac{\widehat{\eta}_k}{\pi_k} \frac{\widehat{\eta}_\ell}{\pi_\ell},$$

$$\widehat{\eta}_k := \widehat{m}(\mathbf{x}_k) + \frac{r_k}{\widehat{p}(\mathbf{x}_k)} \left\{y_k - \widehat{m}(\mathbf{x}_k)\right\}, \quad k \in S.$$

- Special case: SRSWOR

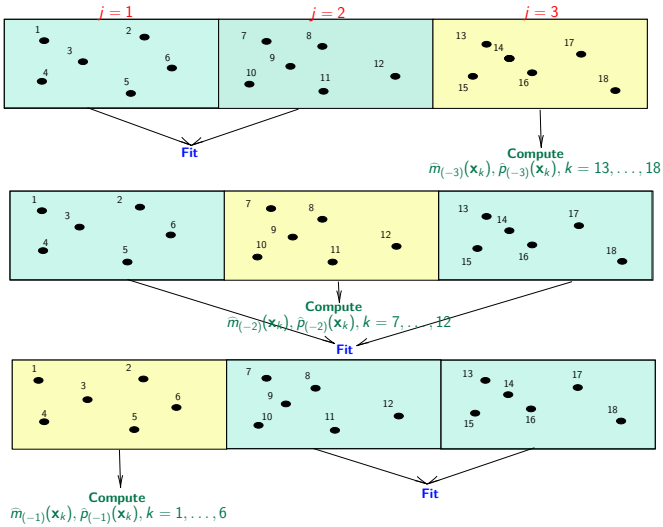
$$\widehat{\mathbb{V}}_{tot}\left\{\widehat{\mu}_{dr}(m, p)\right\} = \left(1 - \frac{n}{N}\right) \frac{s^2_{\widehat{\eta}}}{n}$$

- Issue: Complex ML procedures can overfit noise and end up with artificially small residuals

$\longrightarrow$ Both $\widehat{\mu}_{dr}(\widehat{m}, \widehat{p})$ and $\widehat{\mathbb{V}}_{tot}\left\{\widehat{\mu}_{dr}(\widehat{m}, \widehat{p})\right\}$ may exhibit bias in finite samples.

- Solution: Crossfitting!

Crossfitting with $J = 3$ folds



DR estimation based on crossfitting

- A cross-fitted version of $\widehat{\mu}_{dr}(\widehat{m}, \widehat{p})$ is given by

$$\begin{aligned}\widehat{\mu}_{dr,cf}(\widehat{m}, \widehat{p}) &= \frac{1}{N}\left\{\sum_{k\in S} w_k \widehat{m}_{(-j)}(\mathbf{x}_k) + \sum_{k\in S} w_k \frac{r_k}{\widehat{p}_{(-j)}(\mathbf{x}_k)}\{y_k - \widehat{m}_{(-j)}(\mathbf{x}_k)\}\right\} \\ &= \frac{1}{N}\sum_{k\in S} w_k \widehat{\eta}_{k,cf},\end{aligned}$$

where

$$\widehat{\eta}_{k,cf} = \widehat{m}_{(-j)}(\mathbf{x}_k) + \frac{r_k}{\widehat{p}_{(-j)}(\mathbf{x}_k)}\Big\{y_k - \widehat{m}_{(-j)}(\mathbf{x}_k)\Big\}, \quad \text{for } k \in S.$$

- The residuals based on crossfitting, $y_k - \widehat{m}_{(-j)}(\mathbf{x}_k)$, are called "honest residuals".

DR estimation based on crossfitting

Theorem

We make the following assumption: The sequence of estimators $\{\widehat{m}_v\}_{v\in\mathbb{N}}$ *and* $\{\widehat{p}_v\}_{v\in\mathbb{N}}$ *satisfy the following rate condition:*

$$\lim_{n\to\infty}\sqrt{n}\times\mathbb{E}\left[\frac{1}{|\mathcal{F}_j|}\sum_{k\in\mathcal{F}_j}\left(\widehat{m}_{(-j)}\left(\mathbf{x}_k\right)-m\left(\mathbf{x}_k\right)\right)^2\right]^{1/2}\mathbb{E}\left[\frac{1}{|\mathcal{F}_j|}\sum_{k\in\mathcal{F}_j}\left(\widehat{p}_{(-j)}\left(\mathbf{x}_k\right)-p\left(\mathbf{x}_k\right)\right)^2\right]^{1/2}=0.$$

Then,

$$\widehat{\mu}_{dr,cf}\left(\widehat{m},\widehat{p}\right)-\widehat{\mu}_{dr}\left(m,p\right)=o_p\left(1/\sqrt{n}\right).$$

Both estimators are asymptotically equivalent! $\longrightarrow$ The estimator $\widehat{\mu}_{dr,cf}\left(\widehat{m},\widehat{p}\right)$ inherits all the properties of the oracle estimator $\widehat{\mu}_{dr}\left(m,p\right)$!

DR estimation based on crossfitting

- Root-n consistency:

$$\widehat{\mu}_{dr,cf}\left(\widehat{m},\widehat{p}\right)-\mu=O_p\left(\frac{1}{\sqrt{n}}\right)$$

- If n/N is negligible, an estimator of $\mathbb{V}_{tot}(\widehat{\mu}_{dr,cf}(\widehat{m},\widehat{p}))$, is

$$\widehat{\mathbb{V}}_{tot}\left\{\widehat{\mu}_{dr,cf}(\widehat{m},\widehat{p})\right\}=\frac{1}{N^2}\sum_{k\in S}\sum_{\ell\in S}\frac{(\pi_{k\ell}-\pi_k\pi_\ell)}{\pi_{k\ell}}\frac{\widehat{\eta}_{k,cf}}{\pi_k}\frac{\widehat{\eta}_{k,cf}}{\pi_\ell},$$

where

$$\widehat{\eta}_{k,cf}=\widehat{m}_{(-j)}(\mathbf{x}_k)+\frac{r_k}{\widehat{p}_{(-j)}(\mathbf{x}_k)}\left\{y_k-\widehat{m}_{(-j)}(\mathbf{x}_k)\right\},\quad \text{for } k\in S.$$

- Special case: SRSWOR

$$\widehat{\mathbb{V}}_{tot}\left\{\widehat{\mu}_{dr}(m,p)\right\}=\left(1-\frac{n}{N}\right)\frac{s^2_{\widehat{\eta},cf}}{n}$$

DR estimation based on crossfitting

- Assume the previous assumptions + higher moment conditions on the sampling design. Then,

$$\frac{\widehat{\mathbb{V}}_{tot}\left\{\widehat{\mu}_{dr,cf}(\widehat{m},\widehat{p})\right\}}{\mathbb{V}_{tot}\left\{\widehat{\mu}_{dr,cf}(\widehat{m},\widehat{p})\right\}} \xrightarrow[v\to+\infty]{p} 1.$$

- Asymptotic normality.

$$\frac{\widehat{\mu}_{dr,cf}(\widehat{m},\widehat{p})-\mu}{\sqrt{\widehat{\mathbb{V}}_{tot}\left\{\widehat{\mu}_{dr,cf}(\widehat{m},\widehat{p})\right\}}} \to \mathcal{N}(0,1).$$

Implementation

- The estimator $\widehat{\mu}_{dr,cf}(\widehat{m}, \widehat{p})$ is not written in a convenient form $\longrightarrow$ we seek an estimator of the form

$$\widehat{\mu}_{imp} = \frac{1}{N}\left\{\sum_{k\in S_r} w_k y_k + \sum_{k\in S_m} w_k \widehat{m}(\mathbf{x}_k)\right\}.$$

- Data file after imputation:

Unit	x_1	$\cdots$	x_p	r	w	$ry + (1-r)\widehat{m}$
1	x_{11}	$\cdots$	x_{1p}	1	w_1	y_1
2	x_{21}	$\cdots$	x_{2p}	0	w_2	$\widehat{m}(\mathbf{x}_2)$
3	x_{31}	$\cdots$	x_{3p}	1	w_3	y_3
$\vdots$	$\vdots$	$\ddots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$
n	x_{n1}	$\cdots$	x_{np}	0	w_n	$\widehat{m}(\mathbf{x}_n)$

Implementation

- Obtain the modified imputed values for $k \in S_m$:

$$\widehat{m}^*(\mathbf{x}_k) = \widehat{m}_{(-j)}(\mathbf{x}_k) + \frac{1}{\sum_{\ell \in S_m} w_\ell} \sum_{\ell \in S_r} w_\ell \left\{ \frac{1 - \widehat{p}_{(-j)}(\mathbf{x}_\ell)}{\widehat{p}_{(-j)}(\mathbf{x}_\ell)} \right\} \left\{ y_\ell - \widehat{m}_{(-j)}(\mathbf{x}_\ell) \right\}$$

- Data file after imputation:

Unit	x_1	$\cdots$	x_p	r	w	$ry + (1-r)\widehat{m}^*$
1	x_{11}	$\cdots$	x_{1p}	1	w_1	y_1
2	x_{21}	$\cdots$	x_{2p}	0	w_2	$\widehat{m}^*(\mathbf{x}_2)$
3	x_{31}	$\cdots$	x_{3p}	1	w_3	y_3
$\vdots$	$\vdots$	$\ddots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$
n	x_{n1}	$\cdots$	x_{np}	0	w_n	$\widehat{m}^*(\mathbf{x}_n)$

- As a result, $\frac{1}{N}\left\{\sum_{k \in S_r} w_k y_k + \sum_{k \in S_m} w_k \widehat{m}^*(\mathbf{x}_k)\right\} = \widehat{\mu}_{dr,cf}(\widehat{m}, \widehat{p})$.

A recap...

- **Step 1**: Select a ML procedure for $m(\mathbf{x}_k)$ and a ML procedure for $p(\mathbf{x}_k)$. Optimize the hyperparameters for each ML procedure (e.g., cross-validation).
- **Step 2**: Compute

$$\widehat{\mu}_{dr,cf}(\widehat{m},\widehat{p}) = \frac{1}{N}\left\{\sum_{k\in S} w_k \widehat{m}_{(-j)}(\mathbf{x}_k) + \sum_{k\in S} w_k \frac{r_k}{\widehat{p}_{(-j)}(\mathbf{x}_k)}\{y_k - \widehat{m}_{(-j)}(\mathbf{x}_k)\}\right\}$$

- **Step 3**: Modify the imputed values $\longrightarrow \widehat{m}^*(\mathbf{x}_k)$.
- **Step 4**: Estimate the variance of $\widehat{\mu}_{dr,cf}$ by

$$\widehat{\mathbb{V}}_{tot}(\widehat{\mu}_{dr,cf}(\widehat{m},\widehat{p})) = \frac{1}{N^2}\sum_{k\in S}\sum_{\ell\in S}\frac{(\pi_{k\ell}-\pi_k\pi_\ell)}{\pi_{k\ell}}\frac{\widehat{\eta}_{k,cf}}{\pi_k}\frac{\widehat{\eta}_{\ell,cf}}{\pi_\ell}$$

- **Step 5**: Compute a 95% confidence interval for μ:

$$\widehat{\mu}_{dr,cf}(\widehat{m},\widehat{p}) \pm 1.96\sqrt{\widehat{\mathbb{V}}_{tot}(\widehat{\mu}_{dr,cf}(\widehat{m},\widehat{p}))}$$

Simulation study: Generating the population

- We generated 5 mutually independent predictors $x_1, \ldots, x_5$, according to a standard normal distribution.
- We then repeated $R = 10,000$ iterations of the following process:

  (1) Given the predictors $x_1, \ldots, x_5$, we generated four finite populations with 2 different values of N: 4,000; 16,000.

  One survey variables Y:

  $$Y = 20 \times \mathbb{1}_{x_1 > 0} + 10 \sin(x_2) + 5x_3^2 + \mathcal{N}(0, 1);$$

  (2) A sample S was selected from each population according to simple random sampling without replacement with $n/N = 10\%$
  $\longrightarrow$ n: 400; 1, 600.

Simulation study: Generating nonresponse

(3) In each sample:

The missing values to Y were generated independently from a Bernoulli distribution with probability

$$p(\mathbf{x}_k) = \frac{1}{6}\left\{0.5 + \sin(x_1) + \sin(x_2) + \sin(x_3)\right\}.$$

In each sample, we obtained approximately 50% of missing values.

Simulation study: Estimation through ML

(4) In each sample, we computed $\widehat{\mu}_{dr}(\widehat{m}, \widehat{p})$ and $\widehat{\mu}_{dr,cf}(\widehat{m}, \widehat{p})$ based on $J = 10$ folds:

ML procedure for $m(\mathbf{x}_k)$:

(i) Random forests (RF) with 15 observations in each terminal node and $B = 250$ trees;

(ii) Cubist;

(iii) XGBoost with a learning rate set to 1, a depth set to 5, and $B = 50$ trees.

ML procedure for $p(\mathbf{x}_k)$:

(i) Regression trees with 40 observations in each terminal node

(ii) Random forests (RF) with 20 observations per terminal node and $B = 250$ trees

Simulation study: Monte Carlo Measures

1. Monte Carlo percent relative bias of a point estimator:

$$\mathrm{RB}(\widehat{\mu}) = 100 \times \frac{\mathbb{E}_{MC}(\widehat{\mu} - \mu)}{\mu}.$$

2. Relative efficiency of $\widehat{\mu}$:

$$\mathrm{RE}(\widehat{\mu}) = 100 \times \frac{\mathrm{MSE}_{MC}(\widehat{\mu})}{\mathrm{MSE}_{MC}(\widehat{\mu}_{\mathrm{HT}})}.$$

3. Monte Carlo percent relative bias of variance estimator:

$$\mathrm{RB}\left\{\widehat{\mathbb{V}}(\widehat{\mu})\right\} = 100 \times \frac{\mathbb{E}_{MC}\left\{\widehat{\mathbb{V}}(\widehat{\mu})\right\} - \mathbb{V}_{MC}(\widehat{\mu})}{\mathbb{V}_{MC}(\widehat{\mu})}.$$

4. Monte Carlo coverage rate of 95% normal-based confidence intervals.

Simulation study: Results for the variable Y

	$\widehat{\mu}_{dr}(\widehat{m},\widehat{p})$	$\widehat{\mathbb{V}}(\widehat{\mu}_{dr}(\widehat{m},\widehat{p}))$	$\widehat{\mu}_{dr,cf}(\widehat{m},\widehat{p})$	$\widehat{\mathbb{V}}(\widehat{\mu}_{dr,cf}(\widehat{m},\widehat{p}))$
$[m(\mathbf{x}_k); p(\mathbf{x}_k)]$ = [Cubist; Trees]				
$n = 400$	0.8 (130)	-14.2 (92.6)	0.5 (130)	-2.1 (94.7)
$n = 1,600$	-0.1 (112)	-8.8 (93.8)	-0.1 (113)	-0.8 (94.6)
$[m(\mathbf{x}_k); p(\mathbf{x}_k)]$ = [Cubist; Trees]				
$n = 400$	-0.6 (116)	-14.6 (92.4)	0.2 (129)	-3.5 (94.2)
$n = 1,600$	-0.1 (116)	-11.3 (93.7)	0.0 (120)	0.0 (95.0)

Table 1: Monte Carlo statistics associated with point and variance estimator for the survey variable Y

Simulation study: Results for the variable Y

	$\widehat{\mu}_{dr}(\widehat{m},\widehat{p})$	$\widehat{\mathbb{V}}(\widehat{\mu}_{dr}(\widehat{m},\widehat{p}))$	$\widehat{\mu}_{dr,cf}(\widehat{m},\widehat{p})$	$\widehat{\mathbb{V}}(\widehat{\mu}_{dr,cf}(\widehat{m},\widehat{p}))$
$[m(\mathbf{x}_k); p(\mathbf{x}_k)] = [\text{RF; RF}]$				
$n = 400$	0.1 (158)	-20.2 (92.6)	0.1 (160)	-0.5 (95.3)
$n = 1,600$	0.1 (152)	-23.0 (91.8)	0.1 (152)	-6.4 (94.9)
$[m(\mathbf{x}_k); p(\mathbf{x}_k)] = [\text{XGBoost; Trees}]$				
$n = 400$	-0.2 (201)	-52.0 (82.4)	0.2 (228)	-2.1 (94.7)
$n = 1,600$	-0.1 (195)	-50.4 (83.9)	0.0 (218)	0.4 (95.0)

Table 2: Monte Carlo statistics associated with point and variance estimator for the survey variable Y

What happens in the case of a poor architecture?

	$\widehat{\mu}_{dr,cf}(\widehat{m},\widehat{p})$	$\widehat{\mathbb{V}}(\widehat{\mu}_{dr,cf}(\widehat{m},\widehat{p}))$	MSE($\widehat{m}$)	MSE($\widehat{p}$)
	$[m(\mathbf{x}_k); p(\mathbf{x}_k)] = [\text{Cubist}; \text{XGBoost}]$			
$n = 200$	-1.2 (583)	50.8 (98.3)	0.29	0.108
$n = 400$	-0.5 (514)	52.5 (98.4)	0.18	0.108
$n = 800$	-0.3 (484)	51.7 (98.3)	0.13	0.103
$n = 1,600$	-0.2 (446)	48.3 (98.3)	0.09	0.093

Table 3: Results for the survey variable Y

Final Remarks

- Instead of a single ML procedure for estimating $p(\mathbf{x}_k)$ and $m(\mathbf{x}_k)$, we can use an aggregation procedure for either or both of them.
- Extension to random imputation for categorical variables: Use of a rejective imputation procedure
- Extension to finite population parameters that can be expressed as a solution of an estimating equation (e.g., distribution function, quantiles): Under investigation
- Unified theory for model-assisted estimation based on ML procedures: Under investigation.
- Small area estimation with ML: Under investigation