Semiparametric and nonparametric calibration estimators in cluster sampling by use of penalty functions

Pius Nderitu Kihara
Department of Statistics and Actuarial Sciences, Technical University of Kenya, P.O. BOX 52428-00200 Nairobi, Kenya
pius.kihara@yahoo.com

Available online at: www.isca.in, www.isca.me
Received 4th January 2018, revised 28th March 2018, accepted 10th April 2018

Abstract

The application of nonparametric model calibration estimators in multistage survey sampling has been studied by several authors with the cluster level auxiliary information assumed completely available for each cluster. The reasoning behind model calibration is that if the calibration constraints are satisfied by the auxiliary variable, then it is expected that the fitted values of the variable of interest should satisfy such constraints too. In this paper, we have considered a case of auxiliary information present at two levels. We derive estimators by treating the calibration problems at both levels as optimization problems and solving them by the method of penalty functions. We have shown that the estimators obtained are robust since they do not fail in the event the model is misspecified for the data.

Keywords: Optimization problem, semiparametric model, nonparametric model, model calibration, penalty function.

Introduction

The nonparametric and semiparametric modeling techniques have become popular due to the failings of parametric modeling when a model is misspecified. Given a sample $S$ of $n$ triple of observations $(Z_i, x_i, y_i), i = 1, 2, \ldots, n$ from some population $U$ of size say $N$, of interest is to find an estimator for $E(y_i) = g(x_i, Z_i)$ of a missing population value. Once the missing values are imputed, an estimate of the total of the population of the dependent variable $Y$ can then be obtained. Breidt et al\(^1\) considered a super population regression model, $E(x)$

given by

$$E(\hat{y}_i) = g(x_i, Z_i) = \mu(x_i) + Z_i\beta$$

and used a sample estimate of the form $\hat{g}_i = \hat{\mu}(x_i) + Z_i\hat{\beta}$ with $\hat{\mu}(x_i)$ obtained by local polynomial nonparametric method. Accordingly, they obtained the following estimator for population total

$$y_{\text{reg}} = \sum_U \hat{g}_i + \sum_s \frac{y_i - \hat{g}_i}{\pi_i}$$

They found that the estimator shares some desirable properties with the fully parametric regression estimators. It is location and scale invariant, and it is internally calibrated for both the parametric and the nonparametric components, in the sense that $\hat{X}_{\text{reg}} = \sum_U x_i$ and $\hat{Z}_{\text{reg}} = \sum_U Z_i$. The estimator was shown to be design consistent with the rate $\sqrt{n}$, in the sense that

$$y_{\text{reg}} = \sum_U y_i + O_p \left(\frac{1}{\sqrt{n}}\right).$$

Kihara et al\(^2\) extended the work of Breidt et al\(^1\) to include model calibration in cluster sampling with auxiliary information available at both element and cluster levels and missing values fitted nonparametrically and semiparametrically by use of penalized splines. The work by Kihara\(^3\) considered calibration problem as an optimization problem where missing values were fitted parametrically. Further work by Kihara\(^4\) considered the calibration problem, in one stage sampling, as an optimization problem with missing values fitted nonparametrically and semiparametrically.

In this study, the work by Kihara et al\(^2\) is extended by treating the two levels calibration problems, in cluster sampling, as constrained nonlinear optimization problems which we convert to unconstrained optimization problems. We solve the resulting problems by penalty function method to obtain the weights (at both cluster and cluster element levels) assigned to sample observations from some chi-square distance measures.

Two Level Model Calibration in cluster Sampling

Consider a population $U$ partitioned into $M$ clusters each of size $N_i$ and let $C$ be the population of the clusters. For all clusters included in the sample $s$, two independent vectors, $x_i$ and $z_i$ are available where $z_i$ is a categorical vector. For simplicity, we let $x_i$ be a scalar. At stage one, a sample $s$ of size $m$ consisting of clusters, is selected from $C$ as per a fixed
design \( p_i(.). \) Let \( \pi_i = p(i \in s) \) and \( \pi_j = p(i, j \in s) \) be the marginal and joint cluster inclusion probabilities respectively. From each of the sampled cluster \( i \in s \), a sample \( s_i \) and size \( n_i \) consisting of cluster elements is selected as per a fixed design \( p_i(.) \) with the respective marginal and joint element inclusion probabilities as \( \pi_{ki} = p(k \in s_i / i \in s) \) and \( \pi_{kl} = p(k, l \in s_i / i \in s) \). We assume invariance and independence of the second stage design. Let \( t_i = g(x_i, Z_i) + \varepsilon_i, i = 1, 2, \ldots, M \) , where the smooth function \( g(x_i, Z_i) \) is the fitted model mean for the \( i \)th cluster total. For \( \hat{\pi}_i \) defined for every \( i \), we propose a semiparametric estimator \( \hat{\pi}_i \). We derive a model calibrated estimator of cluster total for every \( i \), we propose a semiparametric estimator for \( \pi_i \) as

\[
\hat{\pi}_i = \frac{\sum_{k \in s_i} w_k \hat{g}_k}{\sum_{k \in s_i} w_k} - N_i \]

(4)

where \( \hat{\pi}_i \) is the fitted model mean for the \( i \)th cluster total. We define the semiparametric estimator for \( \pi_i \) as \( \hat{\pi}_i \) obtained from the sampled clusters.

Now, consider the case where there is also auxiliary information known at element level such that for each element in the \( i \)th cluster, a nonparametric variable \( x_{ik} \) and a categorical vector \( Z_{ik} \) are available. Suppose that not all element values of the variable of interest in a given cluster are available and have to be imputed. We derive a model calibrated estimator of cluster total. We define the semiparametric estimator for \( \pi_i \) as

\[
\hat{\pi}_i = \frac{\sum_{k \in s_i} w_k \hat{g}_k}{\sum_{k \in s_i} w_k} - N_i \]

(3)

Where: \( \hat{\mu}(x_{ik}) \) and \( x_{ik} \) are defined for every element \( k \) in the cluster \( C_j \). For simplicity, we write \( \hat{\mu}_i \) for \( \hat{\mu}(x_{ik}) \). We propose a model calibrated estimator of cluster total to be

\[
\hat{t}_i = \sum_{k \in s_i} w_k \hat{g}_k \]

(4)

with the weights \( w_k \) derived in such a way that the chi square distance measure below is minimized as discussed by Deville and Sarndal.\(^3\)

\[
\Phi_y = \sum_{k \in s_i} \left( \frac{w_k - d_k}{q_k d_k} \right)^2 \]

(5)

The distance measure is minimized subject to the constraints \( \sum_{k \in s_i} w_k = N_i \) and \( \sum_{k \in s_i} w_k \hat{g}_k = \sum_{k \in s_i} \hat{g}_k \) proposed by Wu and Sitter.\(^6\) We have the optimization problem below similar to the one of Kihara.\(^4\)

\[
\text{minimize } \Phi_y = \sum_{k \in s_i} \left( \frac{w_k - d_k}{q_k d_k} \right)^2 \text{ subject to}
\]

\[
\begin{align*}
L_1 (w_i) &= \sum_{k \in s_i} w_k \hat{g}_k - \sum_{k \in C_i} \hat{g}_k = 0 \quad \text{and} \\
L_2 (w_i) &= \sum_{k \in s_i} w_k = N_i = 0
\end{align*}
\]

(6)

We construct an unconstrained problem as given below. See Rao.\(^7\)

\[
\phi(w, r_a) = \sum_{k \in s_i} \left[ \frac{w_k - d_k}{q_k d_k} \right] + H(r_a) \left[ \sum_{k \in s_i} \hat{g}_k - \sum_{k \in C_i} \hat{g}_k \right]^2 + H(r_a) \left[ \sum_{k \in s_i} w_k - N_i \right]^2
\]

(7)

Now, \( H(r_a) \) is a function of some penalty \( r_a \).

Differentiating (7) partially with respect to \( w_k \) we get

\[
\phi'(w, r_a) = \frac{2(w_k - d_k)}{q_k d_k} + 2H(r_a) \left[ \sum_{k \in s_i} \hat{g}_k - \sum_{k \in C_i} \hat{g}_k \right] + 2H(r_a) \left[ \sum_{k \in s_i} w_k - N_i \right]
\]

(8)

Equating (8) to zero and solving for \( w_k \) we have

\[
w_k = \frac{\sum_{j \in s_k} y_{jk} \hat{g}_j + \sum_{j \in C_k} \hat{g}_j}{1 + H(r_a) \left[ (\hat{g}_k)^2 + 1 \right] q_k d_k}
\]

(9)

Thus, a semiparametric estimator of the cluster total is given as

\[
\hat{t}_i = \sum_{k \in s_i} w_k \hat{y}_k = \frac{\sum_{k \in s_i} y_{ik} d_k}{1 + H(r_a) \left[ (\hat{g}_k)^2 + 1 \right] q_k d_k}
\]

(10)

Now, having estimated the cluster totals, we then derive a population total estimator using the estimated cluster totals and the auxiliary information available at cluster level. With \( \hat{g}_k \) and \( x_{ik} \) defined for every \( i \in C \), we propose a semiparametric model calibrated population total estimator as

\[
\hat{y}_{ip} = \sum_{i \in s} w_i \hat{t}_i
\]

(11)

with \( w_i \) obtained in such a way that the chi square distance measure below is minimized.

\[
\Phi = \sum_{i \in s} \left( \frac{y_i - d_i}{q_i d_i} \right)^2
\]

(12)
The distance measure is minimized Subject to the constraints 
\[ \sum_{i \in S} w_i = M \quad \text{and} \quad \sum_{i \in S} w_i \hat{g}_i = \sum_{i \in C} \hat{g}_i. \]
Again, \( d_i = \pi^{-1} \) and \( q_i \) are some known positive constants uncorrelated with \( d_i \). We therefore have the optimization problem

\[
\begin{align*}
\min \quad \Phi &= \sum_{i \in S} \frac{(w_i - d_i)^2}{q_i d_i} \quad \text{subject to} \\
& \quad \begin{cases}
L_1 (w) = \sum_{i \in S} w_i \hat{g}_i - \sum_{i \in C} \hat{g}_i = 0 \quad \text{and} \\
L_2 (w) = \sum_{i \in S} w_i - M = 0
\end{cases}
\end{align*}
\]

(13)

We convert (13) to an unconstrained optimization problem below

\[
\phi(w, r_b) = \sum_{i \in S} \frac{(w_i - d_i)^2}{q_i d_i} + H(r_b) \left[ \sum_{i \in S} w_i \hat{g}_i - \sum_{i \in C} \hat{g}_i \right] + H(r_b) \left[ \sum_{i \in S} w_i - M \right]^2
\]

(14)

Where: \( r_b \) is some penalty.

Differentiating (14) partially with respect to \( w_i \) we get

\[
\phi'(w, r_b) = \frac{2(w_i - d_i)}{q_i d_i} + 2H(r_b) \left[ \sum_{j \in i} w_j \hat{g}_j - \sum_{j \in C} \hat{g}_j \right] + 2H(r_b) \left[ \sum_{i \in S} w_i - M \right]
\]

(15)

We equate (15) to zero and solve for \( w_i \) to obtain the following.

\[
w_i = \frac{d_i - H(r_b)q_i d_i \left[ \sum_{j \in i} w_j \hat{g}_j + 1 - \sum_{j \in C} \hat{g}_j \right]}{1 + H(r_b) \left( \sum_{j \in C} \hat{g}_j \right)^2 + 1 q_i d_i}
\]

(16)

Now we have a semiparametric estimator of the population total obtained as

\[
y_{ap} = \sum_{j \in S} w_j \hat{t}_j = \sum_{j \in S} \frac{\hat{t}_j d_i}{1 + H(r_b) \left( \sum_{j \in C} \hat{g}_j \right)^2 + 1 q_i d_i}
\]

(17)

\[
H(r_b)q_i d_i \left[ \sum_{j \in i} w_j \hat{g}_j + 1 - \sum_{j \in C} \hat{g}_j \right] - \sum_{j \in C} \hat{g}_j \left( \sum_{j \in C} \hat{g}_j \right)^2 + 1 q_i d_i
\]

When the vectors \( Z_i = Z_{ik} = 0 \), then \( \hat{g}(x_{ik}, z_{ik}) = \hat{\mu}(x_{ik}) \) and \( \hat{g}(x_i, z_i) = \hat{\mu}(x_i) \). If we let \( \hat{\mu}(x_{ik}) = \hat{\mu}_{ik} \) and \( \hat{\mu}(x_i) = \hat{\mu}_i \), we have a nonparametric model calibrated estimator for cluster total as

\[
i_{m} = \sum_{j \in S} w_j y_{ij} = \sum_{j \in S} \left[ y_{ij} d_i + H(r_b) \left( \hat{\mu}_i \right)^2 + 1 q_i d_i \right]
\]

(18)

\[
H(r_b)q_i d_i y_{ij} \left[ \sum_{j \in i} w_j \hat{\mu}_j + 1 - \sum_{j \in C} \hat{\mu}_j \right] - \sum_{j \in C} \hat{\mu}_j \left( \sum_{j \in C} \hat{\mu}_j \right)^2 + 1 q_i d_i
\]

and the nonparametric population total estimator becomes

\[
y_{ap} = \sum_{j \in S} w_j \hat{t}_m = \sum_{j \in S} \frac{\hat{t}_m d_i}{1 + H(r_b) \left( \sum_{j \in C} \hat{\mu}_j \right)^2 + 1 q_i d_i}
\]

(19)

\[
H(r_b)q_i d_i \hat{t}_m \left[ \sum_{j \in i} w_j \hat{\mu}_j + 1 - \sum_{j \in C} \hat{\mu}_j \right] - \sum_{j \in C} \hat{\mu}_j \left( \sum_{j \in C} \hat{\mu}_j \right)^2 + 1 q_i d_i
\]

For a semiparametric case, to obtain the within cluster weights \( w_{ik}, (k = 1, 2, ..., n_i) \), we solve the penalty function (7) as an unconstrained minimization problem. Starting with some initial guess for \( w_{ik} \) and \( r_a \), we repetitively improve on the guess until optimal values are obtained. Given that our constraints are equality constraints, our initial guess for \( w_{ik} \) is not required to be feasible as explained in Kihara. We make use of the Newton method discussed in Rao.

Let \( W_i = \left\{ w_{i1}, w_{i2}, ..., w_{im} \right\} \) be our set of weights. We wish to derive \( W^* \) so that

\[
\phi(W_i^*) = \left[ \phi'(w_{i1}, r_a), ..., \phi'(w_{im}, r_a) \right] = 0
\]

(20)

We let \( W_i \) be the initial approximation of \( W_i^* \) so that \( W_i = W_{il} + V_i \). By Taylor’s series expansion of \( \phi(W_i^*) \) we get

\[
\phi(W_i^*) = \phi(W_{il} + V_i) = \phi(W_{il}) + J_{W_{il}} V_i + .......
\]

(21)

If we ignore the higher order terms in (21) and set \( \phi(W_i^*) = 0 \), we get

\[
\phi(W_{il}) + J_{W_{il}} V_i = 0
\]

(22)
The matrix $J_{w_{ij}}$ consists of the second order derivatives of the penalty function (7) evaluated at $W_{ij}$. In general, the $J_{w_{ij}}$ matrix is a $n_i$ by $n_i$ matrix. Let $k$ and $j$ denote the rows and columns respectively.

Then, $J_{w_{ij}}$ has the elements $\frac{2}{q_{ik}d_i} + 2H(r_j)((\hat{\phi}_i)^2 + 1)$ in the diagonal and the elements $2H(r_j)((\hat{\phi}_i + 1)$ elsewhere. If $J_{w_{ij}}$ is invertible, then, from the linear equations (22) we have

$$V_i = J_{w_{ii}}^{-1}\vartheta(W_{ii})$$

(23)

The iterative procedure below is used in finding the enhanced approximations of $W^*_i$.

$$W_{i(i+1)} = W_i - J_{w_{ii}}^{-1}\vartheta(W_{ii})$$

(24)

The sequence of the points $W_{i1}, W_{i2}, \ldots, W_{i(i+1)}$ will eventually converge to the actual solution $W^*_i$.

Let $W^*_i$ be the minimum value of $W_i^*$ calculated for a given penalty $r_i$, we calculate a sequence of minimum points $W^*_{i1}, W^*_{i2}, \ldots, W^*_{i(a+1)}$ for the penalties $r_1, r_2, \ldots, r_{a+1}$ until $W^*_{ia} = W^*_{i(a+1)}$ or $\phi(W_i, r_a) = \phi(W_i, r_{a+1})$ to a given degree of accuracy. The penalty values are such that the initial value $r_1 > 0$ and $r_{a+1} = c r_a$, where $c < 1$. $H(r_a) \rightarrow \infty$ as $r_a \rightarrow 0$.

In nonparametric case, $\hat{\mu}_i$ replaces $\hat{\phi}_i$ so that $J_{w_{ii}}$ matrix is then a $n_i$ by $n_i$ matrix with diagonal elements $\frac{2}{q_{ik}d_i} + 2H(r_j)((\hat{\mu}_i)^2 + 1)$ and the elements $2H(r_a)(\hat{\mu}_i + 1)$ elsewhere.

We next obtain the cluster level weights $w_i$, $(i = 1, 2, \ldots, m)$. Considering the semiparametric case, we solve the penalty function (14) as an unconstrained minimization problem. Let the set of weights be $W = \{w_1, w_2, \ldots, w_m\}$. We require $W^*$ such that

$$\vartheta(W^*) = [\phi^*(w_1, r_b) \ldots, \phi^*(w_m, r_b)] = 0$$

(25)

We let $W_i$ be the initial estimate of $W^*$ so that $W^* = W_i + V$. The Taylor’s series expansion of $\vartheta(W^*)$ now gives

$$\vartheta(W^*) = \vartheta(W_i + V) = \vartheta(W_i) + J_{w_{ii}}V + \ldots$$

(26)

Again, we neglect the higher order terms in (26) and set $\vartheta(W^*) = 0$ to arrive at

$$\vartheta(W_i) + J_{w_{ii}}V = 0$$

(27)

Where: $J_{w_{ii}}$ is the $m$ by $m$ matrix of second order derivatives of the penalty function (14) evaluated at $W_i$. Let $i$ and $j$ be the row and column counters respectively. The matrix $J_{w_{ij}}$ has elements $\frac{2}{q_{ik}d_i} + 2H(r_j)((\hat{\mu}_i)^2 + 1)$ in the main diagonal and the elements $2H(r_j)(\hat{\mu}_i + 1)$ elsewhere. For nonparametric case, the matrix has $\frac{2}{q_{ik}d_i} + 2H(r_j)((\hat{\mu}_i)^2 + 1)$ as diagonal elements and the elements $2H(r_j)(\hat{\mu}_i + 1)$ elsewhere.

We now have the iterative procedure below to find the improved estimates of $W^*$.

$$W_{i+1} = W_i - J_{w_{ii}}^{-1}\vartheta(W_i)$$

(28)

Letting $W^*_b$ be the minimum value of $W^*$ calculated for a given penalty $r_b$, we again calculate a sequence of minimum points $W^*_1, W^*_2, \ldots, W^*_b$ for the penalties $r_1, r_2, \ldots, r_{b+1}$ until $W^*_b = W^*_{b+1}$ or $\phi(w, r_b) = \phi(w, r_{b+1})$ to a specified degree of accuracy. The penalty values for $r_b$ may be set in similar manner as $r_a$ described above.

**Local Polynomial Method of Fitting the Missing Values**

The aim in local polynomial regression is to minimize the degree $q$ polynomial

$$\sum_{j=1}^{n}\left[y_j - \beta_0 - \beta_1(x_j - x_i) - \beta_2(x_j - x_i)^2 \right]K(x_j - x_i)$$

(29)

with respect to $\beta = (\beta_0, \beta_1, \ldots, \beta_p)$ where $\beta_0$ estimates $\mu(x_i) = \mu$, while $\beta_1, \ldots, \beta_p$ estimates higher order derivatives of
The kernel function $K(.)$ is discussed in Simonof. From the local polynomial smoother, the nonparametric fit of the cluster totals can be obtained as

$$\hat{\mu}_j = S_n^T \hat{T}_n$$ (30)

Where: $S_n^T = e^T (X_n^T \sigma_{ni} X_n)^{-1} X_n^T \sigma_{ni} e = (1,0,...,0)^T$, $\hat{T}_n = (\hat{t}_1, \hat{t}_2, ..., \hat{t}_n)^T$, $\sigma_{ni} = \text{diag}(K((x_i - x_j)/h),...,K((x_n - x_j)/h))$, $h$ is the bandwidth within the $i$th cluster and the matrix $X_{ni}$ has the rows $[1, (x_j - x_i), ..., (x_n - x_i)]$, $j = 1,2,..,n$. A discussion of this is given by Breidt and Opsomer.

In a manner similar to that of Breidt and Opsomer, we obtain a semiparametric fit for cluster totals as

$$\hat{g}_i = S_n^T (Y_i - Z_i^T \hat{\beta}) + Z_i (Z_i^T S_i Z_i)^{-1} Z_i^T S_i \hat{Y}_i$$ (31)

where $S_i = [S_{ni}, i = 1,2,...,n]$, $\hat{\beta} = (Z_i^T S_i Z_i)^{-1} Z_i^T S_i Y_i$ and $Z_i = [Z_i, Z_2, ...,] is the vector of categorical variables.

The nonparametric fit of the elements within clusters is then obtained as

$$\hat{\mu}_{ik} = S_{ik}^T Y_{si}$$ (32)

Where: $S_{ik}^T = e^T_i (X_{nik}^T \sigma_{nik} X_{nik})^{-1} X_{nik}^T \sigma_{nik}^T e_i = (1,0,0,0)^T$, $y_{ji} = (y_{j1}, y_{j2}, ..., y_{jn_i})^T$, $\sigma_{nik} = \text{diag}(K((x_{ij} - x_{ik})/h_i),...,K((x_{in_i} - x_{ik})/h_i))$, $h_i$ is the bandwidth within the $i$th cluster and $X_{nik}$ is a matrix with rows $[1, (x_{ij} - x_{ik}), ..., (x_{in_i} - x_{ik})]^T$, $j = 1, 2, ..., n_i$.

The semiparametric fit of the elements within clusters is similarly obtained as

$$\hat{g}_{ik} = S_{ik}^T (Y_{is} - Z_{is}^T \hat{\beta}) + Z_{ik} (Z_{is}^T S_{is} Z_{is})^{-1} Z_{is}^T S_{is} Y_{si}$$ (33)

Where $S_{is} = [S_{is}, k = 1,2,..,n_i]$, $\hat{\beta}_i = (Z_{is}^T S_{is} Z_{is})^{-1} Z_{is}^T S_{is} Y_{si}$ and $Z_{is} = [Z_{is}, Z_{i2}, ...,] is the vector of categorical variables in the $i$th cluster.

**Results**

We analyze the performance of the derived estimators in comparison to the performance of Horvitz Thompson design estimator $y_{hi} = \sum_{i=1}^{n} \hat{r}_{hi} d_i$ of the population total, where $\hat{r}_{hi} = \sum_{k \in i} y_{ik}$ is the cluster total estimator. In Figures-1 to 4, the sample sizes given are for one stage sampling. That is, sizes $m$ of the samples of clusters. The size $n_i$ of the sub sample within a cluster was set as 0.25 of $m$.

**Semiparametric Estimator Results:** We simulated a population size 300 of independent and identically distributed variable $X$ using uniform (0.1) and a categorical matrix $Z$. For each generated $x_j$ and vector $Z_j$, $N_i = 100$ element values were generated as follows.

$$y_{ik} = \frac{g(x_i, Z_i)}{\sqrt{N_i}} + \frac{e_{ik}}{\sqrt{N_i}}, \{e_{ik}\} \text{iid } N(0,0.1)$$ (34)

Where: $y_{ik}$ is the $k$th element in the $i$th cluster and $g(x_i, Z_i)$, which we simply write $g_i$ is the mean function for the cluster total $t_i$. This generating function is an adaptation to semiparametric modeling of the generating function by Montanari and Ranalli. We considered the linear mean function $Z \beta + 2 + 5x$ and the function $Z \beta + (2 + 5x)^2$ which is quadratic, for auxiliary information at cluster level. For simplicity, within each cluster, the auxiliary information $x_{ik}$ at element level was generated using the linear and quadratic mean functions and working backward in a similar manner as in Kihara to obtain the following.

$$x_{ik} = \frac{y_{ik} - 2 - z_{ik} \beta}{5}$$ (35)

And

$$x_{ik} = \frac{-2 + \sqrt{y_{ik} - z_{ik} \beta}}{5}$$ (36)

Where: $Z_{ik}$ is the matrix $(Z_{i1}, Z_{i2}, Z_{i3}), Z_{i1}$ is a matrix of 1s, $Z_{i2}$ is a matrix of 2s, 3s and 4s, while $Z_{i3}$ is a matrix of 5s, 6s, and 7s. $\beta$ is the matrix $(1, 2, 3)$.

At stage one, samples of clusters of size $m$ were generated by simple random sampling. At stage two, within each of the selected clusters, sub samples of size $n_i$ were generated by simple random sampling. For any combination of sample sizes $m$ and $n_i$, 5 samples were generated at stage one and 10
samples at stage two. We used local polynomial equation (33) in fitting cluster elements and equation (31) in fitting cluster totals and in each case the bandwidths are chosen to be a quarter of the respective range in the data. In our study we used a polynomial of degree 1(local linear) and used the standard kernel defined as $K(v) = 0.75(1-v^2), v \leq 1$.

In estimating cluster totals by the penalty function method, our initial penalty constant for $r_a$ was set at $r_a = 0.00010$. The convergence criteria considered was $W_a^* = W_{a+1}^*$ and $\phi(w, r_a) = \phi(w, r_{a+1})$ to six decimal places. Using the estimated cluster totals, we generated estimates of the population total. Again, we set the initial penalty value for $r_b$ at $r_b = 0.00010$ and the convergence criteria as $W_b^* = W_{b+1}^*$ and $\phi(w, r_b) = \phi(w, r_{b+1})$ to six decimal places. We compared the performance of our estimator $y_{sp}$ with the Horvitz Thompson estimator $y_{ht}$.

### Results on Linear Data:

Looking at table (1), the errors indicate that the performances of both estimators $y_{sp}$ and $y_{ht}$ are indistinguishable, which indicates $y_{sp}$ is as reliable as the popular design estimator $y_{ht}$. Convergence at both stage 1 and stage 2 occurs at the initial values of the penalties. From Figure-1, it can be seen that, from the ratio $\text{variance}(y_{sp}) / \text{variance}(y_{ht})$, $y_{sp}$ is a bit more variable than Horvitz Thompson estimator $y_{ht}$.

### Table-1: Results of $y_{sp}$ on Linear Data.

<table>
<thead>
<tr>
<th>Sample serial number</th>
<th>Sample sizes $m$ and $n$</th>
<th>$y_t$</th>
<th>$y_{sp}$</th>
<th>$y_{ht}$</th>
<th>$y_t - y_{sp}$</th>
<th>$y_t - y_{ht}$</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>100 and 50</td>
<td>10637.07767</td>
<td>10655.64312</td>
<td>10713.64452</td>
<td>-18.56545</td>
<td>-76.56685</td>
</tr>
<tr>
<td></td>
<td>2</td>
<td>10637.07767</td>
<td>10591.80901</td>
<td>10551.51204</td>
<td>45.26866</td>
<td>85.56685</td>
</tr>
<tr>
<td></td>
<td>3</td>
<td>10637.07767</td>
<td>10776.77585</td>
<td>10587.80962</td>
<td>-139.69818</td>
<td>49.26805</td>
</tr>
<tr>
<td></td>
<td>4</td>
<td>10637.07767</td>
<td>10657.06451</td>
<td>10579.51231</td>
<td>-19.98684</td>
<td>57.56536</td>
</tr>
<tr>
<td></td>
<td>5</td>
<td>10637.07767</td>
<td>10616.70892</td>
<td>10722.72513</td>
<td>20.36875</td>
<td>-85.64746</td>
</tr>
</tbody>
</table>

### Figure-1: Fraction of $\text{variance}(y_{sp}) / \text{variance}(y_{ht})$ on Linear Data
Results on Quadratic Data: Looking at table (2), again the errors in the estimation indicate that the performances of $y_{sp}$ and $y_{ht}$ are indistinguishable. This serves to show robustness of the estimator $y_{sp}$ which is in fact a misspecified model for quadratic data. In figure (2), we see that, from the ratio $\frac{\text{variance}(y_{sp})}{\text{variance}(y_{ht})}$, the estimator $y_{sp}$ has bigger variance than $y_{ht}$. This is expected since the data is from a quadratic function, while $y_{sp}$ uses a local linear function in fitting the values.

Nonparametric Estimator Results: Using R software program, and using uniform (0, 1), a population of the variable $x$ was simulated. Using the auxiliary variable $x$, two populations for the dependent random variable $y$ were generated as $y = 2 + 5x$ and $y = (2 + 5x)^2$. We used local polynomial equation (30) to fit cluster totals and equation (32) to fit element values within a cluster. The cluster element values were generated as
\[ y_{ik} = \frac{\mu(x_i)}{N_i} + \frac{\epsilon_{ik}}{\sqrt{N_i}}, \{\epsilon_{ik}\} \text{iid } N(0,1) \]  

Table-2: Results of $y_{sp}$ on Quadratic Data.

<table>
<thead>
<tr>
<th>sample serial number</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
</tr>
</thead>
<tbody>
<tr>
<td>sample size $m$ and $n$</td>
<td>100 and 50</td>
<td>100 and 50</td>
<td>100 and 50</td>
<td>100 and 50</td>
<td>100 and 50</td>
</tr>
<tr>
<td>$y_{t}$</td>
<td>16054.39204</td>
<td>16054.39204</td>
<td>16054.39204</td>
<td>16054.39204</td>
<td>16054.39204</td>
</tr>
<tr>
<td>$y_{sp}$</td>
<td>16077.78872</td>
<td>16389.73764</td>
<td>15525.60252</td>
<td>15845.52393</td>
<td>15936.92781</td>
</tr>
<tr>
<td>$y_{ht}$</td>
<td>15706.03516</td>
<td>16389.95682</td>
<td>16259.14548</td>
<td>16174.82744</td>
<td>16081.07106</td>
</tr>
<tr>
<td>$y_{t} - y_{sp}$</td>
<td>-23.39668</td>
<td>-335.3456</td>
<td>521.78952</td>
<td>208.86811</td>
<td>117.46423</td>
</tr>
<tr>
<td>$y_{t} - y_{ht}$</td>
<td>347.60332</td>
<td>-335.56478</td>
<td>-204.75344</td>
<td>-120.4354</td>
<td>-26.67902</td>
</tr>
<tr>
<td>$r_a$</td>
<td>0.00010</td>
<td>0.00010</td>
<td>0.00010</td>
<td>0.00010</td>
<td>0.00010</td>
</tr>
<tr>
<td>$r_b$</td>
<td>0.00010</td>
<td>0.00010</td>
<td>0.00010</td>
<td>0.00010</td>
<td>0.00010</td>
</tr>
</tbody>
</table>

Figure-2: Fraction of $\frac{\text{variance}(y_{sp})}{\text{variance}(y_{ht})}$ on Quadratic Data.
The respective auxiliary information was regenerated as shown below for the linear and quadratic mean functions.

\[ x_k = \frac{y_{ik} - 2}{5} \]  
\[ x_k = \sqrt{\frac{y_{ik} - 2}{5}} \]  

Results on Linear Data: From Table-3, both estimators \( y_{np} \) and \( y_{ht} \) have small errors and consistently, \( y_{np} \) has the smaller error margins. This can be explained by the fact that the data is linear which implies that \( y_{np} \) is correctly specified for the data. From Figure-3, we see that the ratio \( \text{variance}(y_{np}) / \text{variance}(y_{ht}) \) increases as the sample size grows up to a constant of about 0.37. Thus, the variance for \( y_{np} \) is consistently lower than that of \( y_{ht} \). This can be explained by the fact that \( y_{np} \) is correctly specified for the given data.

Table-3: Results of \( y_{np} \) on Linear Data.

<table>
<thead>
<tr>
<th>Sample serial number</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
</tr>
</thead>
<tbody>
<tr>
<td>Sample size ( m ) and ( n )</td>
<td>100 and 50</td>
<td>100 and 50</td>
<td>100 and 50</td>
<td>100 and 50</td>
<td>100 and 50</td>
</tr>
<tr>
<td>( y_{t} )</td>
<td>1344.531793</td>
<td>1344.531793</td>
<td>1344.531793</td>
<td>1344.531793</td>
<td>1344.531793</td>
</tr>
<tr>
<td>( y_{np} )</td>
<td>1345.95725</td>
<td>1340.25334</td>
<td>1327.40832</td>
<td>1349.00000</td>
<td>1350.49969</td>
</tr>
<tr>
<td>( y_{ht} )</td>
<td>1347.04198</td>
<td>1337.97500</td>
<td>1318.56434</td>
<td>1351.24476</td>
<td>1353.21979</td>
</tr>
<tr>
<td>( y_{t} - y_{np} )</td>
<td>-1.425457</td>
<td>4.278453</td>
<td>17.123461</td>
<td>-4.468207</td>
<td>-5.967897</td>
</tr>
<tr>
<td>( y_{t} - y_{ht} )</td>
<td>-2.510187</td>
<td>6.556793</td>
<td>25.967453</td>
<td>-6.712967</td>
<td>-8.687997</td>
</tr>
<tr>
<td>( r_a )</td>
<td>0.00010</td>
<td>0.00010</td>
<td>0.00010</td>
<td>0.00010</td>
<td>0.00010</td>
</tr>
<tr>
<td>( r_b )</td>
<td>0.00010</td>
<td>0.00010</td>
<td>0.00010</td>
<td>0.00010</td>
<td>0.00010</td>
</tr>
</tbody>
</table>

Figure-3: Fraction of \( \text{variance}(y_{np}) / \text{variance}(y_{ht}) \) on Linear Data
Results on Quadratic Data: Looking at Table-4, the errors in estimation indicates that the performances of the estimators $y_{np}$ and $y_{ht}$ are indistinguishable. This points to the robustness of the estimator $y_{np}$ which is misspecified for the quadratic data. The ratio $\frac{\text{variance}(y_{np})}{\text{variance}(y_{ht})}$ seem to tend to a constant as seen in figure (4), though the ratio is a bit wild for small samples. Also, the variance for $y_{np}$ is larger than the variance for $y_{ht}$.

Conclusion

From the results, it is clear that when the nonparametric estimator $y_{np}$ is correctly specified for the data, it is more efficient than the popular Horvitz Thompson design estimator $y_{ht}$ and that $y_{np}$ is only slightly less efficient when it is misspecified for the data. Also, the performance of the semiparametric estimator $y_{sp}$ is indistinguishable from that of the design estimator. We conclude that the semiparametric and nonparametric estimators are robust estimators since they do not fail under misspecification.

In a real world problem where we may not have, or may not be sure that we have all the relevant auxiliary information about a variable, model calibrated estimators would therefore be the estimators of choice. We have shown that in cases where some elements within clusters are unreachable but auxiliary information is available at element level, we can take advantage of this auxiliary information to obtain cluster totals, which are then used in the estimation of population total. We note that if there is a possibility that some clusters may be unreachable, it means there is also the possibility that some cluster elements may be unreachable.

Table-4: Results of $y_{np}$ on Quadratic Data.

<table>
<thead>
<tr>
<th>Sample serial number</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
</tr>
</thead>
<tbody>
<tr>
<td>Sample size $m$ and $n_i$</td>
<td>100 and 50</td>
<td>100 and 50</td>
<td>100 and 50</td>
<td>100 and 50</td>
<td>100 and 50</td>
</tr>
<tr>
<td>$y_{t}$</td>
<td>6702.63067</td>
<td>6702.63067</td>
<td>6702.63067</td>
<td>6702.63067</td>
<td>6702.63067</td>
</tr>
<tr>
<td>$y_{np}$</td>
<td>6989.35523</td>
<td>6579.98771</td>
<td>7013.19892</td>
<td>6677.42846</td>
<td>6716.28391</td>
</tr>
<tr>
<td>$y_{ht}$</td>
<td>6411.78004</td>
<td>6589.61917</td>
<td>6853.44946</td>
<td>6655.73623</td>
<td>6802.89124</td>
</tr>
<tr>
<td>$y_{t} - y_{np}$</td>
<td>-286.72456</td>
<td>122.64296</td>
<td>-310.56825</td>
<td>25.20221</td>
<td>-13.65324</td>
</tr>
<tr>
<td>$y_{t} - y_{ht}$</td>
<td>290.85063</td>
<td>113.0115</td>
<td>-150.81879</td>
<td>46.89444</td>
<td>-100.26057</td>
</tr>
<tr>
<td>$r_a$</td>
<td>0.00010</td>
<td>0.00010</td>
<td>0.00010</td>
<td>0.00010</td>
<td>0.00010</td>
</tr>
<tr>
<td>$r_p$</td>
<td>0.00010</td>
<td>0.00010</td>
<td>0.00010</td>
<td>0.00010</td>
<td>0.00010</td>
</tr>
</tbody>
</table>

Figure-4: Fraction of $\frac{\text{variance}(y_{np})}{\text{variance}(y_{ht})}$ on Quadratic Data.
References


