American Journal of Theoretical and Applied Statistics
2020; 9(4): 162-172
http://www.sciencepublishinggroup.com/j/ajtas
doi: 10.11648/j.ajtas.20200904.20
ISSN: 2326-8999 (Print); ISSN: 2326-9006 (Online)

Population Total Estimation in a Complex Survey by Nonparametric Model Calibration Using Penalty Function Method with Auxiliary Information Known at Cluster Levels

Janiffer Mwende Nthiwa[1, *], Ali Salim Islam[1], Pius Nderitu Kihara[2]

[1]Department of Mathematics, Egerton University, Nakuru, Kenya

[2]Department of Financial and Actuarial Mathematics, Technical University of Kenya, Nairobi, Kenya

Email address:

janienthiwa@yahoo.com (J. M. Nthiwa), asislam54@yahoo.com (A. S. Islam), piuskihara@yahoo.com (P. N. Kihara)

[*]Corresponding author

To cite this article:

Janiffer Mwende Nthiwa, Ali Salim Islam, Pius Nderitu Kihara. Population Total Estimation in a Complex Survey by Nonparametric Model Calibration Using Penalty Function Method with Auxiliary Information Known at Cluster Levels. *American Journal of Theoretical and Applied Statistics*. Vol. 9, No. 4, 2020, pp. 162-172. doi: 10.11648/j.ajtas.20200904.20

Received: July 16, 2020; **Accepted**: August 8, 2020; **Published**: August 19, 2020

Abstract: Nonparametric methods are rich classes of statistical tools that have gained acceptance in most areas of statistics. They have been used in the past by researchers to fit missing values in the presence of auxiliary variables in a sampling survey. Nonparametric methods have been preferred to parametric methods because they make it possible to analyze data, estimate trends and conduct inference without having to fully specify a parametric model for the data. This study, therefore, presents some new attempts in the complex survey through the nonparametric imputation of missing values by the use of both penalized splines and neural networks. More precisely, the study adopted a neural network and penalized splines to estimate the functional relationship between the survey variable and the auxiliary variables. This complex survey data was sampled through a cluster - strata design where a population is divided into clusters which are in turn subdivided into strata. Once missing values have been imputed, this study performs a model calibration with auxiliary information assumed completely available at the cluster level. 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. The population total estimators are derived by treating the calibration problems at cluster level as optimization problems and solving it by the method of penalty functions. A Monte Carlo simulation was run to assess the finite sample performance of the estimators under complex survey data. The efficiency of the estimator's performance was then checked by MSE criterion. A comparison of the penalized spline model calibration and neural network model calibration estimators was done with Horvitz Thompson estimator. From the results, the two nonparametric estimator's performances seem closer to that of Horvitz Thompson estimator and are both unbiased and consistent.

Keywords: Nonparametric Model, Auxiliary Information, Neural Network, Penalized Splines, Optimization Problem

1. Introduction

The concept of auxiliary variables in the present scholarship in statistics denotes independent or predictor variables in a regression analysis. As the name suggests, the variables offer additional information and may be used to improve the estimation of population parameters. As noted by [1], micro-econometric research is frequently performed using data collected by surveys, which may contain auxiliary information for every unit of the population of interest. As can be expected, many of these surveys use complex sampling plans to reduce costs and increase the estimation efficiency for subgroups of the population. Although the word complex survey has been used mostly by researchers to refer to different combinations of sampling plans, however, in this study, complex survey refers to a mixture of both stratified and cluster sampling methods [2].

The processes of estimation of population total and mean starts first with the point estimation of these missing values based on auxiliary variable using either parametric or nonparametric regression estimations. These are then classified according to the nature of the working model used at the estimation stage [11]. As

a result, to obtain the estimator of interest, this study involved two stages of estimation. Stage one was to obtain point estimation of the missing values using penalized splines and neural network based on the auxiliary information at cluster levels. The second stage applied model calibration technique on the fitted cluster total values to obtain the population total as discussed in section 2.0 and 3.0, respectively of this paper. In model calibration, a distance measure defined on some design weights thought to be close to the inclusion probabilities is minimized subject to some calibration constraints imposed on the fitted values of the study variable. In penalty technique, the minimization is usually done by introducing penalty function whose solution gives the optimal design weights to be used in the estimation of population total. This study derived nonparametric model calibration estimators by treating the calibration problem as a nonlinear constrained minimization problem, which in turn transformed into an unconstrained optimization problem using penalty functions.

[7] employs neural networks for imputation with auxiliary information coming from administrative registers. The use of neural networks for model calibration in the study is new and allows for more flexible prediction and straightforward insertion of auxiliary information. A more complex model and generalized calibration procedure using model calibration was proposed by [12]. These scholars considered generalized linear models and nonparametric regression models for the super population model ψ given in the equation below.

$$y_i = h(x_i) + \varepsilon_i \tag{1}$$

where $\{\varepsilon\}_{i=1}^{n}$ is a sequence of independent and identically distributed random variables with $E(\varepsilon_i) = 0$ and $E(\varepsilon_i^2) = \sigma^2$ and $h(x_i)$ is a smooth function which can be estimated by any nonparametric methods like kernel, neural network and penalized splines. Given n pair of sample observations $(x_1, y_1), \ldots, (x_n, y_n)$ from a population of size N, of interest, is the estimator $\hat{h}(x)$ of $h(x) = E_{\psi}(y/x)$. For model calibration, calibration is performed to the population mean of the fitted values $\hat{h}_i(x_i)$ [12]. The study considers a model calibration estimator for population total Y_t given below.

$$\hat{Y} = \sum_{i \varepsilon b} w_i y_i \, . \tag{2}$$

where b is a set of sampled units under a general sampling design while w_i's is the design weights such that for a given metric, are as close as possible in an average sense to the $z_i = \frac{1}{\pi_i}$, where π_i is the inclusion probability. These weights are obtained by minimizing a given distance measure between the w_i's and z_i's subject to some constraints. The chi-squared distance measure to be minimized is as provided in the equation below.

$$\delta_b = \sum_{i \in b} \frac{(w_i - z_i)^2}{q_i z_i} \, . \tag{3}$$

where q_i's are known positive constants uncorrelated with the z_i's, [4] subject to two constraints in equation (4) and (5).

$$\sum_{i \varepsilon b} w_i = N \tag{4}$$

$$\text{and } \sum_{i \varepsilon b} w_i \hat{h}_i = \sum_{i=1}^{N} \hat{h}_i \tag{5}$$

where $\hat{h}_i = \hat{h}_i(x_i)$.

The model calibration method is intended to provide good efficiency when the model is correctly specified, but maintain desirable properties like design consistency when the model is misspecified [10]. This study embarked on a model calibrated rather than internal calibration approach because authors such as [5] showed that model calibrated estimators performed better than internally calibrated estimators.

[6] proposed the use of nonparametric method to obtain the fitted values. In particular, they used neural networks and local polynomial in fitting the missing values for clustered data in one-stage sampling. An extension of this to two-stage sampling using kernel functions was done by [8] to fit the mean functions. Any nonparametric method such as kernel methods can be used to recover the fitted values for the non-sampled units. Such estimators are however challenging to employ in cases of multiple covariates and especiall when data is sparse. Another challenge involves incorporating categorical covariates. It is, therefore, important to consider other techniques of recovering the fitted values like penalized splines and neural networks when data is complex as discussed in the following section 2.0 of this paper.

2. Fitting of Missing Values

In this section, the study considered fitting missing values for a population divided into clusters which are then subdivided into strata. This section considered a case where there is auxiliary information known at Cluster Level only. The cluster total being the variable of interest, it was assumed to be dependent on some auxiliary variable x. The study defined $Q = x_1, x_2 \ldots x_C$ as a population of auxiliary variables of size C with x_i being known at i^{th} cluster. This study further considered population of clusters; F to be partitioned into C clusters, each of size $M_i, i = 1, 2, \ldots, C$.

Further, each cluster contains L_i strata each of size $N_j, j = 1, 2, \ldots, L_i$. Let also y_{ijk} be k^{th} observation in the sample from the j^{th} stratum of i^{th} cluster and x_i be the corresponding auxiliary variable at cluster level. At stage one, a probability sample c of size m_i of clusters is drawn from C according to a fixed design $P_1(\bullet)$ (by simple random sampling), where $P_1(c)$ is the probability of drawing the sample c of size m_i from C. The first order cluster inclusion

probabilities, $P_1(\bullet)$ is $\pi_i = pr(i\varepsilon c) = \sum_{i\varepsilon c} P_1(c)$ and $\pi_{i,t} = pr(i,t\varepsilon c) = \sum_{i,t\varepsilon c} P_1(c)$. The first and the second order probabilities are the probability of including cluster i in the sample and the probability of including clusters i and t in the sample respectively. At stage two, for every sampled cluster $i \in c$, the study chose a sample r_i of elements of size $n_i, i = 1,2,...,c$, where $n_i = n_{i1} + n_{i2} + ... + n_{iL_i}$. Given that $n_{i1}, n_{i2}, ..., n_{iL_i}$ are sample sizes of the sample chosen from L_i strata by proportional allocation with inclusion probabilities $\pi_{k/j/i} = pr(k, j\varepsilon r_i / i\varepsilon c)$ and $\pi_{k,p/j/i} = pr(k, p\varepsilon r_i / i\varepsilon c)$. In this case, the first and second order probabilities are the probability of including element k in the sample r_i of the i^{th} cluster and the probability that unit k and p are both included in the sample r_i respectively.

Let $t_i = h(x_i) + \varepsilon_i;\ i = 1,2,...,C$ be the ith cluster total, where $h(x_i)$ is a smooth function of x. Let also $\hat{t}_c = \left[\hat{t}_i\right]_{i\varepsilon c}$ be the m_i dimension vector of $\hat{t}_i$'s which is obtained in the sample of clusters. This study uses [3] estimator to obtain cluster total estimator given by

$$\hat{t}_i = \sum_{j=1}^{L_i} \left(\frac{N_j}{M_i} \right) \overline{y}_j M_i \tag{6}$$

where

$\overline{y}_j = \sum_{k=1}^{n_{ij}} \frac{y_{ijk}}{n_{ij}}$ is the stratum mean for the j^{th} stratum.

This study considered modelling $\hat{h}(x_i)$ in equation (1) by way of both penalized splines and neural network and performed model calibration on $\hat{h}(x_i)$ in case when auxilliary information is available at cluster level.

For penalized splines, this study considered a population of cluster F of size C for which several values for a random variable were missing at i^{th} cluster. A matrix X_r was considered with rows

$$X_{ri}^T = \left\{1, x_i, ..., x_i^q, \left(x_i - k_1\right)_+^q, ..., \left(x_i - k_l\right)_+^q\right\} \tag{7}$$

for $i\varepsilon F$, q is the degree of the spline, and the $k_1, k_2, ..., k_l$ are the knots, while $(x - k_1)_+ = x - k_1$ if $x > k_1$ and 0 if $x \le k_1$. Further, X_{rc} is the sub matrix of X_r which consists of the rows X_{ri}^T for which the clucter $i\varepsilon c$, $A_\alpha = diag\{0, ..., 0, \alpha, ..., \alpha\}$ with q + 1 zeros on the diagonal followed by l penalty constants α.

The study considered the diagonal matrix of inverse inclusion probabilities $W = diag, i \in F\left\{\frac{1}{\pi_i}\right\}$ and its sample submatrix defined as $W_c = diag, i \in c\left\{\frac{1}{\pi_i}\right\}$.

This study let ψ_1 denote a super population of clusters model. To fit the missing values at the cluster level, the study defined the nonparametric population estimator $E_{\psi_1}(t_i) = \hat{h}_i$. If the fits are based on penalized splines, then the design weighted penalized spline smoother vector at x_i due to Breidt and Opsomer, (2000) is considered as;

$$J_{ps}^T = X_{ri}(X_{rc}{}^T W_c X_{rc} + A\alpha)^{-1} X_{rc}{}^T W_c \tag{8}$$

Equation (8) is such that when applied to the sample t_c it yields the nonparametric fit sample fit at x_i for $E_{\psi_1}(t_i)$ as

$$\hat{h}_{t_i} = J_{ps}^T \hat{t}_c . \tag{9}$$

Secondly, the study proposed neural network structure for the nonparametric fit of the cluster totals defined by;

$$\hat{h}_{t_i} = h(x_i, \breve{\theta}_i) \tag{10}$$

The design consistent estimate $\breve{\theta}_i$ in the above equation (10) was estimated by following same procedure as Montanari and Ranalli (2003) as discussed below. The neural network structure for smooth function, $h(x_i)$ was defined by;

$$h(x_i) = \sum_{q=1}^{Q} \beta_q x_{qi} + \sum_{g=1}^{G} a_g \Phi\left(\sum_{q=1}^{Q} \gamma_{qg} x_{qi} + \gamma_{0g} \right) + a_0 \tag{11}$$

for $i = 1,2,...,C$

In this case G is the number of neurons at the hidden layer; $a_g \in \Re$, for $g = 1,2,...,G$ is the weight of the connection of the $q = 1,2,...,Q$ hidden node with the response variable; $\gamma_{qg} \in \Re$, for $q = 1,2,...,Q$ is the weight attached to the connection between the q^{th}auxiliary variable and the $q = 1,2,...,Q$ hidden node. The scalars a_0 and γ_{0g}, for $g = 1,2,...,G$, represent the activation levels of, respectively, the response variable and the G neurons at the hidden layer.

The parameter G which was the number of neurons at hidden layer was considered fixed. As a result, the set of all network parameters was denoted by θ_i and defined as

$$\theta_i = \left\{\beta_1, ..., \beta_Q, a_0, a_1, ..., a_G, \gamma_{01}, ..., \gamma_{0G}, \gamma_1, ..., \gamma_G\right\} \tag{12}$$

where $\gamma_g = \left(\gamma_{1g}, ..., \gamma_{Qg}\right)'$ for $g = 1,2,...,G$.

To estimate the regression function in equation (11), [12]

proposed obtaining a design consistent estimate of θ_i in equation (12) and therefore, of the regression function at x_i, for the fitted values, for $i = 1,2,...,C$. To that purpose, this study first seek for an estimate $\bar{\theta}_i$ of the model parameters based on the entire finite population after which $\breve{\theta}_i$ was obtained, which was a design consistent estimate of $\bar{\theta}_i$ based on sample data only

Further, the following equation presents the population parameter, $\tilde{\theta}_i$ which was defined as the minimizer in the parameter space Θ of the weighted sum of squared residuals with a weight decay penalty term.

$$\tilde{\theta}_i = \underset{\theta_i \in \Theta}{\arg\min} \left\{ \sum_{i=1}^{C} \frac{1}{\vartheta_i} \left(t_i - h(x_i, \theta_i)\right)^2 + \mu \sum_{l=1}^{r} (\theta_i)_l^2 \right\} \quad (13)$$

where, $\vartheta_i, i = 1,2,...,C$ are known positive weights assumed to be proportional to the variance function $\vartheta(x_i)$, r is the dimension of the vector θ_i and μ is known to be a tuning parameter. Then, $\bar{\theta}_i$ was obtained as the solution of the following equations.

$$\sum_{i=1}^{C} \left\{ \left(t_i - h(x_i, \theta_i)\right) \frac{\partial h(x_i, \theta_i)}{\partial \theta_i} \frac{1}{\vartheta_i} - \frac{1}{C} \theta_i \right\} = 0 \quad (14)$$

The sum on the left-hand side of equation (14) is a population total; then a design consistent estimate $\breve{\theta}_i$ of $\bar{\theta}_i$ is defined as the solution of the design-based sample version of (14) that is the solution of the following equations (15).

$$\sum_{i=1}^{m_i} z_i \left\{ \left(t_i - h(x_i, \theta_i)\right) \frac{\partial h(x_i, \theta_i)}{\partial \theta_i} \frac{1}{\vartheta_i} - \frac{1}{C} \theta_i \right\} = 0 \quad (15)$$

where $z_i = \dfrac{1}{\pi_i}$ is the inverse of inclusion probability.

Using the fitted values in equation (9) and (10) this study proposed two types of model calibrated population total estimator based on Neural Network (y_{NN}) and based on the penalized splines; y_{PS} for auxiliary variable available at cluster level and based on cluster-strata design with a general form as;

$$\hat{y}_{Gen} = \sum_{i \varepsilon c} w_i \hat{t}_i \,. \quad (16)$$

Given the inverse inclusion probability as $z_i = \dfrac{1}{\pi_i}$, the weight w_i for penalized spline estimator was obtained by minimizing the chi-square distance measure in equation (3) subject to the constrains;

$$\sum_{i \varepsilon c} w_i = C \text{ and } \sum_{i \in c} w_i \hat{h}_{t_i} = \sum_{i=1}^{C} \hat{h}_{t_i} \quad (17)$$

The weight w_i for NN estimator was obtained by minimizing the same chi-square distance measure in equation (3) subject to the constrains;

$$\sum_{i \varepsilon c} w_i = C$$

$$\text{and } \sum_{i \in c} w_i h(x_i, \breve{\theta}_i) = \sum_{i \in c} w_i \hat{h}_{t_i} = \sum_{i=1}^{C} \hat{h}_{t_i} \,. \quad (18)$$

3. Penalty Function Method of Obtaining the Weights

The procedure of estimating the optimal weights w_i is done by the penalty function method. This function method transforms the basic constrained optimization problem into an unconstrained optimization problem. In nonparametric model calibration estimation, this study followed the same procedure by [9]. The weight w_i in equation (16) was obtained by minimizing the chi-square distance measure in equation (3) subject to the constraints in equation (17) and (18) for penalized spline and neural network estimators respectively.

In this case, $\hat{h}_{t_i}$ is a nonparametric fit of the missing cluster total at the cluster level. Here, calibration constraint $\sum_{i \in c} w_i \hat{h}_{t_i} - \sum_{i=1}^{C} \hat{h}_{t_i} = 0$ is defined on the fitted values in equations (9) and (10); this is called model calibration. The study then constructed an unconstrained problem as follows.

$$\tau(w, g_c) = \sum_{i \in c} \frac{\left(w_i - z_i\right)^2}{q_i z_i} + \Omega(g_c, v_j(w)), j = 1,2 \quad (19)$$

where $\Omega(g_c, v_j(w))$ is a penalty function. Following same procedure by Rao (1984), equation (19) becames;

$$\tau(w, g_c) = \sum_{i \in c} \frac{\left(w_i - z_i\right)^2}{q_i z_i} + T(g_c) \sum_{j=1}^{2} v_j^q(w) \quad (20)$$

where $T(g_c)$ is some function of the parameter g_c which tends to infinity as g_c tends to zero and also $\sum_{j=1}^{2} v_j^q(w)$ tend to zero. In this study, the penalty terms are chosen such that their values will be small at points away from the constraint boundaries and as the constraint boundaries are approached they will tend to infinity. As a result, the value of τ will also blow up as the constraint boundaries are approached. This study chose the value of $q = 2$. Substituting the constraints $\sum_{j=1}^{2} v_j^q(w)$ with the constraints in equation (17) now in the equation (20) results into;

$$\tau(w,g_c,h_{t_i})=\sum_{i\in c}\frac{(w_i-z_i)^2}{q_i z_i}+T(g_c)\left(\sum_{i=1}^{c}w_i h_{t_i}-\sum_{i=1}^{C}h_{t_i}\right)^2+T(g_c)\left(\sum_{i=1}^{c}w_i-C\right)^2 . \quad (21)$$

Equation (21) above was differentiated partially with respect to w_i to give;

$$\tau^1(w,g_c,\hat{h}_{t_i})=2\frac{(w_i-z_i)}{q_i z_i}+2T(g_c)\hat{h}_{t_i}\left(\sum_{j=1}^{c}w_j\hat{h}_{t_j}-\sum_{j=1}^{C}\hat{h}_{t_j}\right)+2T(g_c)\left(\sum_{i=1}^{c}w_i-C\right). \quad (22)$$

Further, equating (22) to zero and solving for w_i the weight becomes;

$$w_i=\frac{z_i-T(g_c)q_i z_i\left(\sum_{\substack{i=1\\ j\neq i}}^{c}w_j\left[\hat{h}_{t_i}\hat{h}_{t_j}+1\right]-\sum_{j=1}^{C}\left[\hat{h}_{t_i}\hat{h}_{t_j}-1\right]\right)}{1+T(g_c)\left((\hat{h}_{t_i})^2+1)q_i z_i\right)}. \quad (23)$$

A general weighted nonparametric estimator of population total in equation (16) for penalized splines and NN based on cluster strata design when auxiliary information is known at only cluster level is therefore obtained as

$$y_{Gen}=\sum_{i=1}^{c}w_i\hat{h}_{t_i}=\sum_{i=1}^{c}\frac{\hat{h}_{t_i}z_i}{1+T(g_c)\left((\hat{h}_{t_i})^2+1)q_i z_l\right)}-\sum_{i=1}^{c}\frac{T(g_c)q_i z_l\hat{h}_{t_i}\left(\sum_{\substack{i=1\\ j\neq i}}^{c}w_j\left[\hat{h}_{t_i}\hat{h}_{t_j}+1\right]-\sum_{j=1}^{C}\left[\hat{h}_{t_i}\hat{h}_{t_j}-1\right]\right)}{1+T(g_c)\left((\hat{h}_{t_i})^2+1)q_i z_l\right)} \quad (24)$$

To obtain the weights $w_i,(i=1,2,...,c)$ in the equation (24), this study solved the penalty functions (21) as an unconstrained minimization problem using an iterative procedure. The research in this case started with some initial guess for w_i and g_c then iteratively improved on the initial values until optimal values are obtained. The present study, therefore, followed the Newton method of unconstrained optimization, according to [9] as follows.

If $w=w_1,w_2,...,w_{m_i}$ is let to be the set of the weights. Of interest was to obtain W^o such that

$$\Gamma(W^o)=\left[\tau'(w_i,g_c,\hat{h}_{t_i}),...,\tau'(w_c,g_c,\hat{h}_{t_i})\right]'=0 \quad (25)$$

Further if W_i is let to be initial estimate of W^o so that $W^o=W_i+X$. Taylor's series expansion of $\Gamma(W^o)$ gives

$$\Gamma(W^o)=\Gamma(W_i+X)=\Gamma(W_i)+N_{W_i}X+.... \quad (26)$$

By neglecting the higher-order terms in the above equation (26) and setting $\Gamma(W^o)=0$ the study had

$$\Gamma(w_i)+N_{w_i}X=0 \quad (27)$$

where N_{w_i} is a m_i by m_i the matrix of second derivatives of the penalty function equation (21) evaluated at W_i. Let also i and j be the row and column counters respectively with $i=(1,2,...,c)$ rows with $j=(1,2,...,c)$ columns. The matrix N_{w_i} has elements $\frac{2}{q_i z_i}+2T(g_c)\left((\hat{h}_{t_i})^2+1\right)$ in the main diagonal and elements $2T(g_c)\left(\hat{h}_{t_i}\hat{h}_{t_j}+1\right)$ elsewhere.

If N_{w_i} is a nonsingular matrix, then, from the set of linear equations (27) the vector X becomes;

$$X=N_{w_i}^{-1}\Gamma(w_i). \quad (28)$$

The study followed the iterative procedure below inorder to find the improved approximations of W^*

$$W_{i+1}=W_i+X_i=W_i-N_{w_i}^{-1}\Gamma(W_i). \quad (29)$$

The sequence of the points $W_1,W_2,...,W_{i+1}$ eventually converges to the actual solution W^o. When W_c^o is let to be the minimum of W^o obtained for a particular penalty; g_c, the study obtained a sequence of minimum points $W_1^o,W_2^o,...,W_{c+1}^o$. for the penalties $g_1,g_2,...,g_{c+1}$ until $W_c^o=W_{c+1}^o$ or $\tau(w_c,g_c,\hat{h}_{t_i})=\tau(w_c,g_{c+1},\hat{h}_{t_i})$ for some specified accuracy level. This accuracy level may be to a certain decimal points or significance level. Again, the penalty values can be set such that the starting point $g_1>0$

and $g_{c+1} = sg_c$, where $s < 1$. $T(g_c) \to \infty$ as $g_c \to 0$.

4. Empirical Analysis and Discussions

In the simulation study, a population of size 10,000 (200*50=10,000) was simulated from a population structure containing 200 clusters each of size 50. Each cluster had 5 strata of size 10 each. At stage one 10, 20, 30, …, 190 clusters were sampled from the 200 clusters by simple random sampling while at stage two, 5 elements where drawn from each stratum by proportional allocation. This gave sample of 25 elements from each of the sampled clusters. 10 replications per each sample size were generated. For penalized spline method, the number of knots and the Spline penalty were optimally generated.

Using R program, a population of independent and identically distributed variable x was simulated using uniform (0, 1). In this study neural network and penalized splines described in section (2.0) of this paper was used to fit the cluster totals as a quadratic function $t_i = (20 + 6x)^2$ where t_i is the ith cluster total, and x is auxiliary information known at the cluster level. The cluster element values were generated as

$$y_{ijk} = \frac{t_i}{cluster\ size} + error\ term(e_i)/\sqrt{cluster\ size}$$

where y_{ijk} is k^{th} unit in j^{th} stratum of i^{th} cluster

On the other hand, to differentiate one stratum from each other, the errors for the five strata were given as $e_i \in (-0.0001, 0.0001)$ for stratum 1, $e_i \in (-0.0002, 0.0002)$ for stratum 2, $e_i \in (-0.0003, 0.0003)$ for stratum 3, $e_i \in (-0.0004, 0.0004)$ for stratum 4 and $e_i \in (-0.0005, 0.0005)$ for stratum 5.

This study reports on the performance of two estimators and their comparison in performance with that of Horvitz Thompson estimator. The performance of the two nonparametric estimators y_{NN} for neural network and y_{PS} for penalized spline were evaluated using its relative bias R_B and relative efficiency R_E. The relative bias is defined as

$$R_B = \frac{\sum_{r=1}^{R} (y_{est} - Y_{AT})}{R * Y_{AT}}$$

where, y_{est} represents any of estimators $\hat{y}_{NN}$ and $\hat{y}_{PS}$, Y_{AT} is the actual total and R is the replicate number of samples. The relative efficiency was defined as

$$R_E = \frac{MSE(y_{est})}{MSE(y_{HT})}$$

where y_{est} was calculated from the Rth simulated sample and $\hat{y}_{HT}$ is the Horvitz Thompson estimator. Large values of relative efficiencies represent higher efficiency for the design estimator $\hat{y}_{HT}$ over the estimator y_{est} that it's being compared with and vice versa. The $\hat{y}_{HT}$ estimator was defined as $y_{HT} = \sum z_i t_i$ where z_i is the inverse of the inclusion probability given by $z_i = \frac{C}{c}$. The estimator y_{HT} was used as the baseline comparison.

4.1. Normality Test

This study carried out a One-sample Kolmogorov-Smirnov normality test for the three estimators; $\hat{y}_{PS}$, $\hat{y}_{NN}$ and $\hat{y}_{HT}$ before their comparative analysis was done. The p values at α = 0.05 for the quadratic data obtained are as in table 1 below. Figure 1, figure 2 and figure 3 show a sample of graphical representation of normality for the quadratic functions. A p-value greater than the set α = 0.05 significance level means normality is established. The results show that at α = 0.05 the proposed estimators are normal.

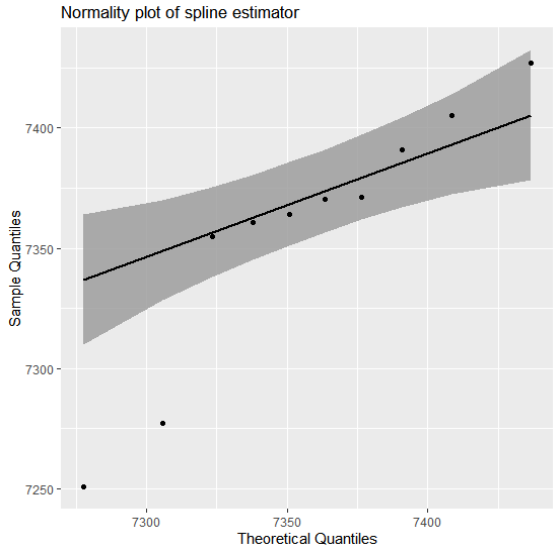

Figure 1. *Normality plot forsplinees estimator.*

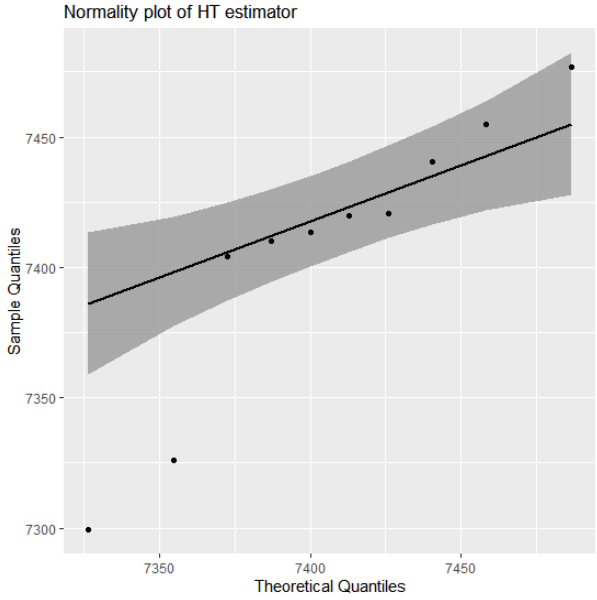

Figure 2. *Normality plot for HT estimator.*

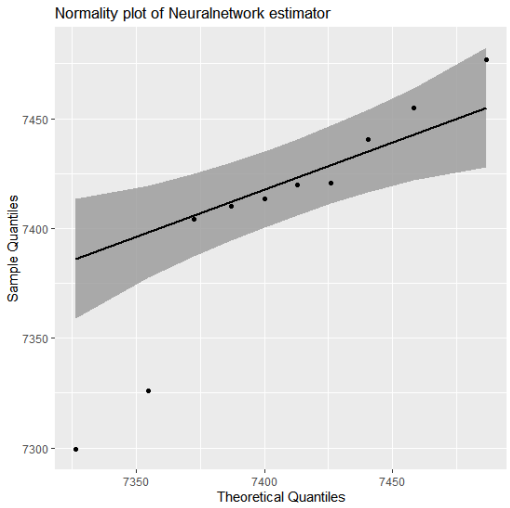

Figure 3. *Normality plot for neural nertwork estimator.*

Table 1. *Normality test.*

	$\hat{y}_{PS}$	$\hat{y}_{HT}$	$\hat{y}_{NN}$
Normality P-values	0.3368	0.3368	0.3368

4.2. Results for Population total Estimates

The table 2 below represents the actual total and the estimates of the penalized splines, neural network and the Horvitz Thompson for 20, 50, 100 and 150 sample sizes. From the results, the estimator y_{PS} seems to give estimates that are very close to those of Horvitz Thompson design estimator for all the four sample numbers. Although neural network estimates are not as close to Horvitz Thompson estimates as penalized splines estimates are, their difference isn't large.

Table 2. *Population total estimates for samples of sizes, 20, 50,100 and 150.*

Replication Number		**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**
y_{AT}	Sample size 20/50/100/150	7378.120	7378.120	7378.120	7378.120	7378.120	7378.120	7378.120	7378.120	7378.120	7378.120
$\hat{y}_{PS}$	20	6923.087	7266.075	8390.214	8337.892	6861.890	7848.353	6853.811	9096.062	6426.442	8387.670
	50	7376.460	6897.797	7168.905	7853.759	6766.208	6802.028	8058.299	7267.496	6864.203	7096.616
	100	7064.701	7353.839	7352.669	7087.797	7793.496	7960.013	7373.334	7144.038	7190.551	7476.643
	150	7119.727	7220.824	7388.137	7302.420	7245.242	7250.961	7692.908	7534.390	7624.514	7490.976
$\hat{y}_{NN}$	20	6968.379	7314.728	8477.491	8432.634	6954.111	7907.010	6952.471	9172.035	6468.121	8461.739
	50	7445.111	6947.934	7216.653	7907.014	6820.390	6847.202	8118.159	7330.350	6909.707	7144.015
	100	7111.888	7403.211	7401.975	7135.046	7846.291	8014.242	7422.843	7191.659	7238.627	7526.937
	150	7167.269	7269.119	7437.667	7351.383	7293.661	7299.453	7744.844	7585.149	7685.760	7541.559
$\hat{y}_{HT}$	20	6923.085	7266.070	8390.205	8337.890	6861.888	7848.350	6853.805	9096.056	6426.438	8387.667
	50	7376.458	6897.795	7168.898	7853.758	6766.205	6802.026	8058.293	7267.494	6864.201	7096.614
	100	7064.699	7353.831	7352.665	7087.789	7793.490	7960.007	7373.333	7144.035	7190.548	7476.641
	150	7119.724	7220.823	7388.135	7302.419	7245.235	7250.955	7692.899	7534.388	7624.510	7490.973

4.3. Results of Variances and Variance Ratios for Various Sample Size

From table 3 below, the variances for the three estimators based on penalized splines, Horvitz Thompson and neural network seem to decrease as the sample size increases. This implies that all the estimators are consistent. For sample sizes; 30, 40, 80, 110, 130,180, and 190 it is seen that penalized spline estimator variance is consistently lower than $\hat{y}_{HT}$ estimator variance. The variance ratio Var($\hat{y}_{PS}$)/ Var($\hat{y}_{HT}$) is slightly greater than one except for sample sizes; 30, 40, 80, 110, 130,180, and 190. On the other hand, the ratios Var($\hat{y}_{NN}$)/ Var($\hat{y}_{HT}$) and Var($\hat{y}_{PS}$)/ Var($\hat{y}_{NN}$), respectively, are all slightly greater and less than one for the 19 replications. This implies that the estimator y_{NN} is slightly highly variant than the both y_{HT} and y_{PS} estimators.

Table 3. *Results of Variances and variance ratios for various sample size.*

Samp size	**10**	**20**	**30**	**40**	**50**	**60**	**70**	**80**	**90**	**100**
$\text{var}(\hat{y}_{PS})$	1,116,735	791,384.3	472,299.4	270,832.1	195,529.9	324895.5	178,372.3	68,206.39	149,070.4	88,285.94
$\text{var}(\hat{y}_{NN})$	1,138,744	805,158.9	481,860.3	274,560.8	199,179.2	329,664.9	180,868.0	69,012.64	151,337.9	89,689.56
$\text{var}(\hat{y}_{HT})$	1,116,731	791,383.1	472,300.3	270,833.1	195,529.3	324,895.1	178,372.3	68,206.81	149,070.3	88,285.64
$\text{var}(\hat{y}_{PS})/\text{var}(\hat{y}_{HT})$	1.000004	1.000002	0.9999982	0.9999963	1.000003	1.000001	1.000000	0.9999939	1.000001	1.000003
$\text{var}(\hat{y}_{NN})/\text{var}(\hat{y}_{HT})$	1.019712	1.017407	1.020241	1.013764	1.018667	1.014681	1.013992	1.011815	1.015212	1.015902
$\text{var}(\hat{y}_{PS})/\text{var}(\hat{y}_{NN})$	0.9806724	0.9828921	0.9801585	0.9864196	0.9816783	0.9855326	0.9862014	0.9883173	0.9850169	0.9843503

Samp size	110	120	130	140	150	160	170	180	190
$\text{var}(\hat{y}_{PS})$	43,555.62	57,071.18	59,375.22	21,950.49	36,370.42	10,787.63	9419.0414911	4761.6977659	2940.8217118
$\text{var}(\hat{y}_{NN})$	44,266.82	57,857.09	60,333.68	22,301.83	37,473.49	10,960.02	9570.5942662	4877.8607639	2987.7138079
$\text{var}(\hat{y}_{HT})$	43,555.74	57,070.94	59,375.44	21,950.33	36,370.16	10,787.50	9419.0151641	4761.7832184	2940.9399535
$\text{var}(\hat{y}_{PS})/\text{var}(\hat{y}_{HT})$	0.9999972	1.000004	0.9999963	1.000007	1.000007	1.000012	1.0000028	0.9999821	0.9999598
$\text{var}(\hat{y}_{NN})/\text{var}(\hat{y}_{HT})$	1.016326	1.013775	1.016139	1.016013	1.030336	1.015992	1.0160929	1.0243769	1.0159044
$\text{var}(\hat{y}_{PS})/\text{var}(\hat{y}_{NN})$	0.9839339	0.9864164	0.9841140	0.9842465	0.9705641	0.9842713	0.9841647	0.9761857	0.9843050

Results in figure 4 represent variance for the three estimators. The variances in this figure seem to decrease as the sample size increases implying that all the estimators are consistent. Figure 5 represents the variance ratio; Var($\hat{y}_{PS}$)/ Var($\hat{y}_{HT}$). This figure shows that the ratio was concentrated around one implying that both estimators are equally variant. Figure 6 and figure 7 shows the variance ratio of Var($\hat{y}_{NN}$)/ Var($\hat{y}_{HT}$) and Var($\hat{y}_{PS}$)/ Var($\hat{y}_{NN}$) respectively. Figure 6 shows that the ratio points concentrated slightly above one while for figure 7, slightly below one implying that $\hat{y}_{NN}$ is slightly more variant than both $\hat{y}_{HT}$ and $\hat{y}_{PS}$ estimators.

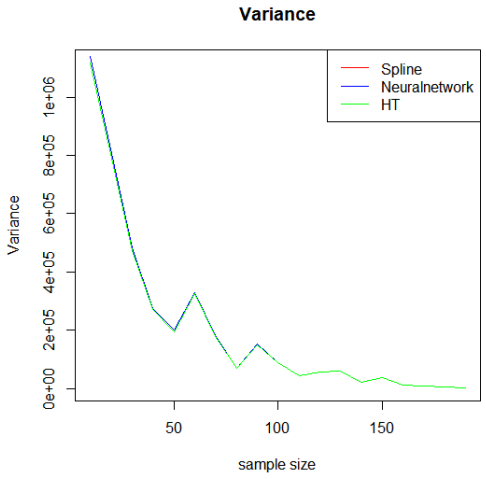

Figure 4. *Variance for spline, HT and neural nertwork estimators.*

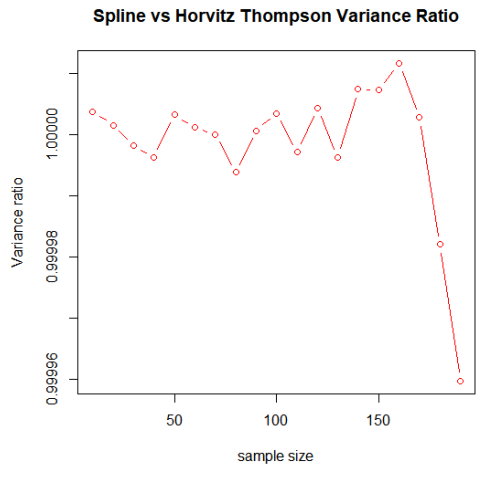

Figure 5. *Variance ratio for spline and HT estimators.*

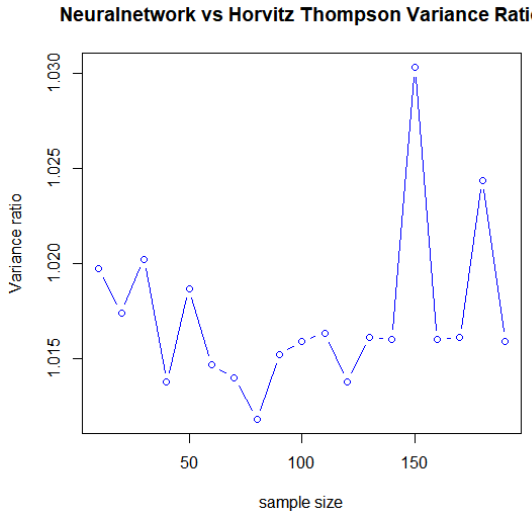

Figure 6. *Variance ratio for neural nertwork and HT estimators.*

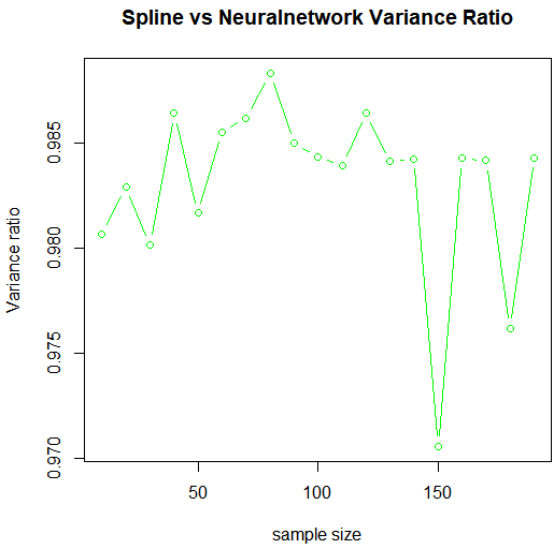

Figure 7. *Variance ratio for spline and neural nertwork estimators.*

4.4. Relative Bias

The following table 4 shows the values of relative biases. The results from the table show that the Relative biases for the three estimates are minimal given that the population totals were in thousands and this point to unbiasedness. On the other hand, comparing the penalized spline estimator with its corresponding Horvitz Thompson estimators, the difference is not significant, and they both have reduced bias

than the neural network estimator.

Table 4. *Bias and Relative Biases for three estimators.*

	$\hat{y}_{PS}$	$\hat{y}_{HT}$	$\hat{y}_{NN}$
Relative bias	0.002228853	0.002228399	0.01006926

4.5. Results on Mean Squared Errors for Various Sample Sizes

This section presents both mean square error and relative efficiencies of the three estimators, y_{PS} based on a penalized spline, y_{HT} based on Horvitz Thompson and y_{NN} base on neural network and their respective graphs. The MSE and relative efficiency for different estimators are summarized in table 5 and table 6, respectively

4.5.1. Mean Squared Errors for the Three Estimators

Generally, the estimator with a smaller MSE is regarded as the most efficient one. From table 5, MSEs of $\hat{y}_{PS}$ and $\hat{y}_{HT}$ seems to be smaller than that of $\hat{y}_{NN}$ in some samples. However, from other sample sizes of 30, 50, 60, 140, 170 and 180 the $\hat{y}_{NN}$ seems to have reduced MSE than both $\hat{y}_{PS}$ and $\hat{y}_{HT}$.

Table 5. *Mean squared errors.*

sample size	**10**	**20**	**30**	**40**	**50**	**60**	**70**	**80**	**90**	**100**
MSE $\hat{y}_{PS}$	1,370,169	859,520.8	580,612.3	272,088.1	222,080.3	334,603.4	179,408.7	71,437.53	163,451.4	88288.461294
MSE $\hat{y}_{NN}$	1,590,084	915,882.8	558,692.0	274,834.0	211,162.1	332,052.8	188,068.0	80,527.14	180,894.9	92306.082951
MSE $\hat{y}_{HT}$	1,370,161	859,5174	580,615.0	272,089.3	222,080.5	334,603.6	179,408.5	71,437.52	163,450.2	88288.144870

sample size	**110**	**120**	**130**	**140**	**150**	**160**	**170**	**180**	**190**
MSE $\hat{y}_{PS}$	44,149.37	57,281.52	62,500.49	27,073.59	36,449.45	11,280.14	11,057.85	5626.4989832	3378.4032344
MSE $\hat{y}_{NN}$	49,747.63	62,026.60	71,536.54	22,811.91	41,009.77	16,125.52	9,646.633	5292.7885287	3796.9954252
MSE $\hat{y}_{HT}$	44,149.34	57,281.20	62,500.33	27,073.94	36,4491.2	11,279.88	11,058.01	5626.7551753	3378.6445113

4.5.2. Relative Efficiency (MSE Ratios) for Various Sample Size

Table 6 on relative efficiency examines the efficiency of the various estimators, i.e. the, MSE $\hat{y}_{PS}$ /MSE $\hat{y}_{HT}$, MSE $\hat{y}_{NN}$ /MSE $\hat{y}_{HT}$ and MSE $\hat{y}_{PS}$ /MSE $\hat{y}_{NN}$. There does not appear to be a noticeable difference in the performances of both $\hat{y}_{PS}$ and $\hat{y}_{NN}$ in comparison to $\hat{y}_{HT}$. In some instances, both $\hat{y}_{PS}$ and $\hat{y}_{NN}$ has smaller error margins than $\hat{y}_{HT}$, while in other samples, $\hat{y}_{HT}$ has lower error margins than both nonparametric estimators. For example, for the sample sizes 30, 40, 50, 60, 140, 170, 180 and 190 the estimator $\hat{y}_{PS}$ have high efficiency than $\hat{y}_{HT}$, while in samples sizes,30, 50, 60, 140, 170 and 180 estimator $\hat{y}_{PS}$ has high efficiency than $\hat{y}_{HT}$. This lack of noticeable difference in the performances of the three estimators may point to the robustness of the estimators $\hat{y}_{PS}$ and $\hat{y}_{NN}$. On the other hand, in 6 of the sample replications with sample sizes 30, 50, 60, 140, 170 and 180 $\hat{y}_{NN}$ seems to be more efficient than $\hat{y}_{PS}$. This shows that both estimators may not be different in efficiency.

Table 6. *Relative efficiency (MSE ratios) for various sample size.*

sample size	**10**	**20**	**30**	**40**	**50**	**60**	**70**	**80**	**90**	**100**
MSE $\hat{y}_{PS}$ /MSE $\hat{y}_{HT}$	1.000006	1.000004	0.9999953	0.9999954	0.9999988	0.9999994	1.000001	1.000000	1.000007	1.000004
MSE $\hat{y}_{NN}$ /MSE $\hat{y}_{HT}$	1.160509	1.065578	0.9622417	1.010087	0.9508358	0.9923766	1.048267	1.127239	1.106728	1.045509
MSE $\hat{y}_{PS}$ /MSE $\hat{y}_{NN}$	0.8616962	0.9384616	1.039235	0.9900089	1.051705	1.007681	0.9539565	0.8871237	0.9035710	0.956475

sample size	**110**	**120**	**130**	**140**	**150**	**160**	**170**	**180**	**190**
MSE $\hat{y}_{PS}$ /MSE $\hat{y}_{HT}$	1.000001	1.000006	1.000003	0.9999868	1.000009	1.000023	0.9999854	0.9999545	0.9999286
MSE $\hat{y}_{NN}$ /MSE $\hat{y}_{HT}$	1.126804	1.082844	1.144579	0.8425780	1.125124	1.429583	0.8723659	0.9406467	1.1238221
MSE $\hat{y}_{PS}$ /MSE $\hat{y}_{NN}$	0.8874667	0.9234994	0.8736862	1.186818	0.8887994	0.6995209	1.146291	1.0630500	0.8897570

4.5.3. MSE and MSE Ratio Graphs for Various Sample Sizes

Results in figure 8 show that the MSE for the three estimators decreases as the sample size increases, this point to the consistency of the three estimators. Figure 9 and figure 10 shows relative efficiency of the proposed nonparametric estimators ratio MSE $\hat{y}_{PS}$ /MSE $\hat{y}_{HT}$ and MSE $\hat{y}_{NN}$ /MSE $\hat{y}_{HT}$ respectively. The ratio for figure 9 is mostly concentrated at a point slightly below one and a point around one for figure 10. This implies that both nonparametric estimators are efficiently competing with the design estimator y_{HT}. Figure 11 shows relative efficiency of the proposed nonparametric estimators ratio MSE $\hat{y}_{PS}$ /MSE $\hat{y}_{NN}$. This ratio is mostly concentrated at a point around one as well implying that both estimators may be equally efficient.

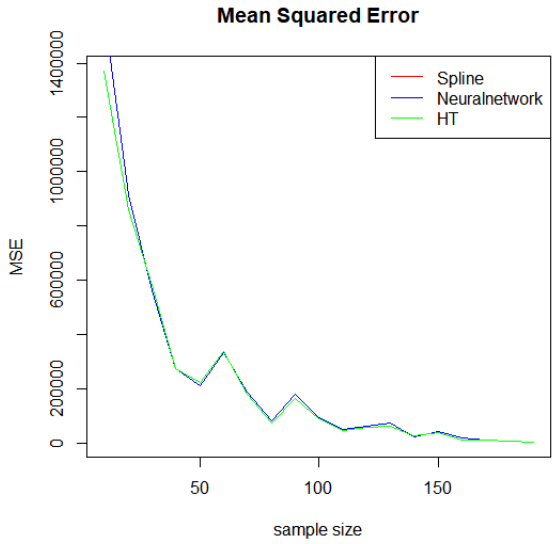

Figure 8. *MSE for spline, HT and neuralnertwork estimators.*

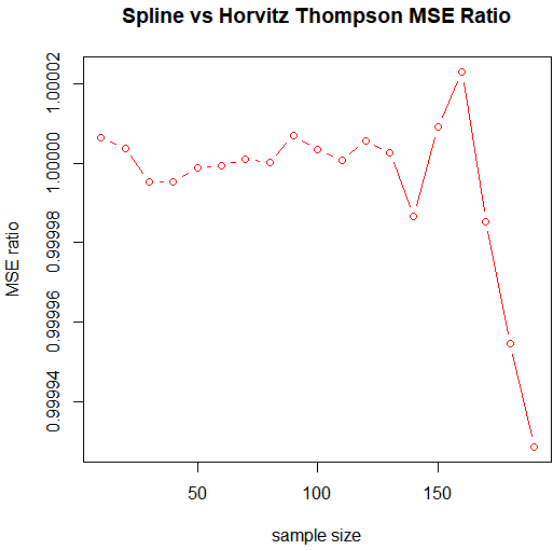

Figure 9. *MSE ratio for spline and HT estimators.*

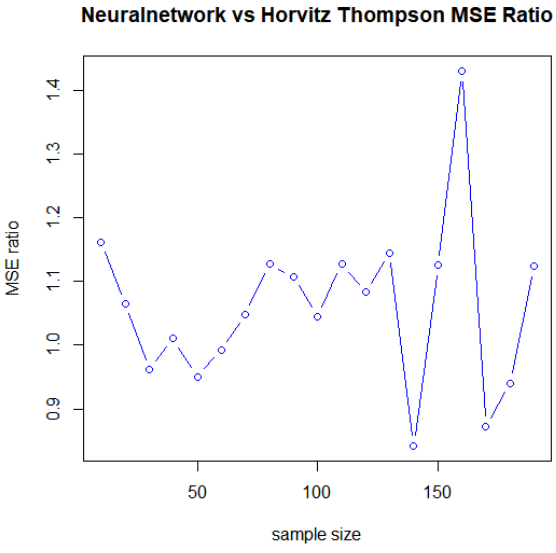

Figure 10. *MSE ratio for neural nertwork and HT estimators.*

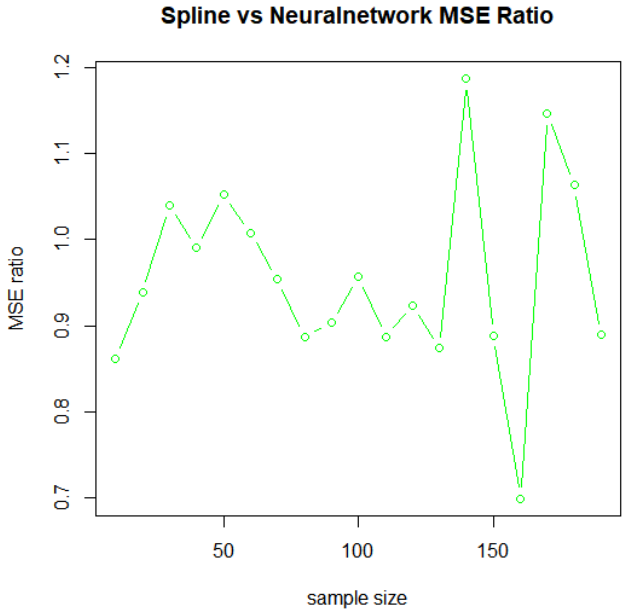

Figure 11. *MSE ratio for spline and neural nertwork estimators.*

5. Conclusion

This study concludes that both estimators $\hat{y}_{PS}$ and $\hat{y}_{NN}$ are competitively efficient estimators since they interchangeably yield smaller errors in estimation in comparison with the design estimator; $\hat{y}_{HT}$. Both nonparametric estimators $\hat{y}_{PS}$ and $\hat{y}_{NN}$ appear not to have a noticeable difference in their performances in comparison to $\hat{y}_{HT}$. As pointed out in setion 4.5, both $\hat{y}_{PS}$ and $\hat{y}_{NN}$ have smaller error margins than $\hat{y}_{HT}$ in some instances, while in other samples, $\hat{y}_{HT}$ has smaller error margins than both nonparametric estimators. This lack of noticeable difference in the performances of the three estimators may point to the robustness of the estimators $\hat{y}_{PS}$ and $\hat{y}_{NN}$. The results in sections 4.3 and 4.4 show the two nonparametric model calibrated estimators are consistent and unbiased, respectively. The design estimator; $\hat{y}_{HT}$ is considered to be a very reliable design estimator and therefore, this study concludes that the two nonparametric model calibrated estimators are also quite reliable as well.

This study can be applied to a real-world problem. In sampling, there are cases whereby some information may be missing due to non-sampling, non-response or even due to non-observed. Still, there is relevant auxiliary information about a variable at the cluster level. In such instances, this study recommends model calibrated estimators to be the estimators of choice. This study has shown that in cases where there are missing values at the cluster level but the auxiliary information is available at such level, then, an advantage can be taken of this auxiliary information to obtain cluster totals, which are then used in the estimation of population total.

References

[1] Breidt, F. J. and Opsomer, J. D. (2000). Local Polynomial Regression Estimation in Survey Sampling. *Annals of Statistics*, 28: 1026 - 1053.

[2] Clair, l. (2016). Nonparametric kernel estimation methods using Complex survey data, PhD thesis, *mcmaster university, Main St. West, Hamilton Ontario*.

[3] Cochran, W. G.. (1977). Sampling techniques (3rd ed.)., New york: John Wiley & sons.

[4] Deville J. C. and Sarndal C. E. (1992). Calibration Estimators in Survey Sampling. *Journal of the American Statistical Association,* 87: 376-382.

[5] Kihara, P. N. (2012). Estimation of Finite Population Total in the Face of Missing Values Using Model Calibration and Model Assistance on Semiparametric and Nonparametric Models. PhD thesis, JKUAT.

[6] Montanari, G. E. and Ranalli, S. (2003). Nonparametric Model Calibration Estimation in Survey Sampling. *Journal of Official Statistics*, 2: 1-40.

[7] Nordbotten, S. (1996). Neural Network imputation applied to the Norwegian 1990 population census data. *Journal of Official Statistics*, 12: 385-401.

[8] Otieno *et al*., (2007). Nonparametric Model Assisted Model Calibrated Estimation in Two Stage Survey Sampling. The East African Journal of Statistics, 3: 261-281.

[9] Rao, S. S. (1984). Optimization Theory and Applications. *Wiley Eastern Limited* Sahar, Z. Z. (2012). Model-based methods for robust finite population inference in the presence of external information. *The University of Michigan.*

[10] Sahar, Z. Z. (2012). Model-based methods for robust finite population inference in the presence

[11] of external information. *The University of Michigan.*

[12] Sarndal, C. E., Swensson B. and Wretman J. (1992). Model Assisted Survey Sampling. *Springer-Verlag*, New York.

[13] Wu, C. and Sitter, R. R. (2001). A Model Calibration Approach to Using Complete Auxiliary Information from Survey Data. *Journal of American Statistical Association*, 96: 185-193.