NON-PARAMETRIC ESTIMATOR FOR A FINITE POPULATION TOTAL UNDER STRATIFIED SAMPLING INCORPORATING A HYBRID OF DATA REFLECTION AND TRANSFORMATION TECHNIQUES

NICHOLAS MUGAMBI

A Project Submitted in Partial Fulfillment of Requirements for Conferment of the Degree of Master of Science in Applied Statistics of Meru University of Science and Technology

2023

NON-PARAMETRIC ESTIMATOR FOR A FINITE POPULATION TOTAL UNDER STRATIFIED SAMPLING INCORPORATING A HYBRID OF DATA REFLECTION AND TRANSFORMATION TECHNIQUES

NICHOLAS MUGAMBI

MASTER OF SCIENCE IN APPLIED STATISTICS

A Project Submitted in Partial Fulfillment of Requirements for Conferment of the Degree of Master of Science in Applied Statistics of Meru University of Science and Technology

2023

DECLARATION

This project is my original work and has not been presented for a degree in any other Institution

Signature…… **Date……..**

Nicholas Mugambi

SC401/0001/18

Supervisors

This project has been submitted with our approval as the University Supervisor

Signature…….. **Date……..**

Professor Romanus Odhiambo, Ph.D

Meru University of Science and Technology

Signature……. **Date……**

Jacob Oketch Okungu

Meru University of Science and Technology

DEDICATION

I wish to dedicate this work to my late grandfather Stanley Mbobua and my family for their love and moral support. You all mean a lot to me.

ACKNOWLEDGEMENT

First of all, let me thank the Almighty God for without His grace and blessings this study would not have been successful. I have experienced His guidance day by day and I will keep on trusting Him in everything.

Words cannot express my deepest gratitude to my supervisors Prof. Romanus Odhiambo and Jacob Oketch Okungu who made this work possible. Their guidance, provision of reference materials, suggestions and invaluable patience, feedback and advice, availability for consultations benefited me much in the completion and success of this project.

I also appreciate the defense committee who made my defense to be enjoyable for their comments and who generously provided knowledge and expertise towards the research.

I would like to express my sincere gratitude to my family as a whole for their continuous support and understanding when undertaking my research. Their belief in me has kept my spirits and motivation high during this process. Their prayers for me have sustained me this far.

May our gracious God bless you all.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

ABSTRACT

Statisticians use survey sampling methods in the estimation of population parameters of interest. This field has received increased demand due to the reliable statistic they produce. Information is extracted from the samples and used to make inferences about the population and used for planning purposes. This information is collected either by survey sampling or census. However, census is an expensive and tedious method to carry out in the estimation process thus preferring survey sampling in estimation. In survey sampling, estimation can be either parametric or nonparametric. In nonparametric, estimation of finite population total divides into the sampled and non-sampled parts. Estimation of the sampled part is quite easy thus the problem reduces to the estimation of non-sampled part. Different approaches have been used by statisticians in the estimation of the non-sampled part. These approaches have however relied on the use kernel smoothers and has been known to suffer the problem of boundary bias. In this study, a nonparametric estimator for a finite population total that addresses this drawback of kernel smoothers is proposed. The properties of this estimator were studied empirically in order to determine its efficiency. The estimator was applied to a simulated data and comparative analysis was done using R statistical software version i386 4.0.3 and the results of the bias were confirmed. The performance of the proposed estimator was tested and compared against the design-based Horvitz-Thompson estimator, the model-based approach proposed by Dorfman and the ratio estimator. The proposed estimator was developed by modifying the Nadaraya-Watson kernel estimator using two boundary bias reducing techniques. The bias, variance and Mean Squared Error of the estimator were studied theoretically and applied to an empirical study out using simulated data from linear, quadratic and exponential mean functions. Both the unconditional and conditional properties of the estimators under the three mean functions were investigated. The proposed estimator outperformed the ratio estimator, Horvitz-Thompson estimator and the estimator due to Dorfman in quadratic and exponential mean functions. This is evident from the small biases and mean squared error values obtained. For the linear mean function, the ratio estimator gave the best estimates because it is (BLUE). Therefore, the proposed nonparametric estimator for a finite population total was developed, the asymptotic properties were studied and comparative analysis done using simulated data. From the results obtained, the proposed estimator was found to give smaller biases and therefore can be recommended for bias correction at the boundary. The proposed estimator in this study is based on stratified sampling, thus a study using cluster sampling is recommended to compare the performance of the estimator and further research to improve the estimator to work for all theoretical data variables.

CHAPTER ONE

1.1 Introduction

This chapter focuses on the background of the study, statement of the problem and the objectives. It also brings to our view the significance and scope of the study.

1.2 Background of the Study

The intensions of surveys are not only in estimating population target parameters, but also in the estimation of subpopulation characteristics. These subpopulations are commonly referred to as domains or areas. The term small area is used to denote a small demographic group, for example a small group characterized by social economic status or age-sex/ethnicity group. Thus, in sample survey, researchers extract information from samples and use such information in making inference about some population quantities such as the mean, proportion or totals. The collection of information can be done either by the use of sampling methods or census. However, census is a tedious and expensive method as it entails complete enumeration of individuals or units contained in a population. Therefore, statisticians rely on the use of sampling methods which involve the selection of a sample from a population of interest and use information obtained from such samples to get estimators of the whole population (Cochran, 1977).

In survey sampling, the estimation of finite population quantities of interest such as the proportions, averages or totals can be done using nonparametric regression method. Nonparametric regression method was introduced early on in the studies by (Nadaraya, 1964) and (Watson, 1964). According to (Dorfman, 1992), estimators obtained using nonparametric regression method are considered to be more flexible and robust as compared to the estimators based on parametric regression methods. In sample survey, researchers use auxiliary information in estimating finite population parameters of interest. However, the use of auxiliary information in estimation of parameters is a key problem in sample surveys. To address this problem,

statisticians assume a working super population model to describe the relationship between the auxiliary variable X and the study variable Y (Dorfman, 1992). This super population working model is used in the prediction of the non-sampled part of the population (Sanchez-Borrego & Rueda, 2009)

In sampling survey, areas with small sample sizes are commonly referred to as small areas. This field of small area estimation in survey sampling has gained more attention over the years because of greatly increasing demand for reliable small area statistics from both public and private sectors. Small area statistics are very important in Government agencies around the world as they are used in program development, allocation of various funds and in regional and city planning and more so in other policies of the government, for example Bureau of labor Statistics and Census Bureau in the United States, Ministry of Social Development of Chile, National Administrative Department of Statistics Colombia and many others. Additionally, these area statistics are also important in industries and private sector policy making of businesses since they rely on local social-economic conditions. The demands are attributed to the significant advances in statistical data processing and powerful statistical methods for the analysis of local area data (Fay & Herriot, 1979; Kriegler & Berk, 2010).

Small area estimation has been accepted widely in recent years and this has led to the development of different models, studied and applied. According to (Pfeffermann, 2013), (Rao, 2003; 2013), and Rao and (Molina, 2015) there has occurred several advances in methodologies used in small area estimation from its start to the present. Small area estimation methodology can be categorized into two types, design-Based approach, for example synthetic, composite and direct estimators and model-based approach for example, unit-level model and area-level model. In Design-Based techniques, there is no existence of models. The estimates of direct survey for small domains here gives out large standard errors (Rao, 2003; 2013). Therefore, in the construction of estimators, it becomes necessary to borrow strength from related areas through

linking models based on auxiliary data such as administrative archives and census data to find more accurate estimates for a given area. On the other hand, in Model-Based or Model-Dependent techniques, the inference is made according to the underlying model. In the attempt to compare the two approaches, (Pfeffermann, 2002; 2013) in his study recommends the use of Model-Based method since the estimators in this approach are more accurate and predictions of non-sampled areas can be done. In addition, Design-Based method has got roles in Model-Based approach as it is the input data for the model under Area-Level model. It can also be used to assess the model predictors.

In this study, we consider a nonparametric estimation of finite population total with a hybrid method working on the basis of stratified sampling. Generally, it is good for every small area to be included in the sample. In order to achieve this, we need to have independent stratification within the small areas. Nonparametric regression approach has been preferred mostly by researchers due to their standing out results which other approaches have failed. The motivations behind nonparametric regression according to (Härdle, 1994) include; It provides a tool for finding spurious observations by studying the influence of isolated points. Secondly, it gives predictions of observations yet to be made without reference to a fixed parametric model. Thirdly, it provides a versatile method for exploring a general relationship between two variables and lastly, it constitutes a flexible method for substituting for missing values or interpolating between adjacent independent random variable values.

1.3 Statement of the Problem

According to (Dorfman, 1992), researchers are basically faced with the challenge of estimating population parameters of interest such as totals, means or even the distribution function of the population. To enhance the performance of the estimators, the available auxiliary information is used. Estimation of population total in nonparametric regression divides into the sampled part and non-sampled part. Estimation of the sampled part is easy thus the problem reduces to the

estimation of non-sampled part. The use of auxiliary information to the non-sampled part of the population could lead to misspecifications during the choice of models. Different nonparametric methods have been employed by researchers in the estimation. However, estimation of the quantity of interest in nonparametric regression relies on the use of kernel smoothers which is an approach used to develop robust estimators (Breidt & Opsomer, 2009). These kernel smoothers are known to suffer the problem of boundary bias. Therefore, a way of improving the nonparametric regression estimation is necessary. (Lang'at et al, 2019) used a modified NW estimator under reflection technique to address the boundary problem associated with the estimator. In this study, a nonparametric regression estimator for a finite population total that addresses the boundary bias problem is proposed composing a composite of data transformation and reflection techniques.

1.4 Objectives of the Study

1.4.1 General Objective

To propose a nonparametric estimator for a finite population total under stratified random sampling incorporating a hybrid of data transformation and reflection techniques.

1.4.2 Specific Objectives

The study was anchored on the following four objectives;

i. To propose a nonparametric estimator for a finite population total based on employing a hybrid of data transformation and data reflection.
ii. To study the properties of the proposed estimator
iii. To apply the proposed estimator to a simulated data.
iv. To compare the performance of the proposed estimator with the Horvitz-Thompson estimator, ratio estimator and the estimator due to (Dorfman, 1992)

1.5 Significance of the Study.

The study of nonparametric regression estimation in sample survey is an important field in statistics. It provides the techniques for estimating the population parameters by the use of samples obtained from the population of interest. Therefore, this study is of great importance towards development the mathematical and statistical knowledge in survey sampling. Thus, the results of this study are pivotal in government agencies in the implementation of policies and planning of various sectors of the economy and even the private sector in their prediction.

In their work, statisticians usually endeavor to have unbiased estimators of the target parameters. However, from the previous studies done by researchers over the years, the attainment of such estimators still remains a challenge to be catered for. Majority of the obtained estimators suffer from the problem of boundary biases. In this study, an estimator for finite population total with minimal bias is proposed and does not suffer from the boundary problems significantly. Therefore, this is in turn an important study to statisticians as they are ever keen on having estimators that are of high precision.

1.6 Scope of the Study

This research project presents a nonparametric regression estimator for a finite population total with a modified kernel smoother that incorporates a hybrid of data transformation and reflection techniques under model-based framework. The asymptotic properties of the proposed estimator have been studied empirically and analytically. The estimator was applied to a simulated data and the performance of the proposed estimator was compared against the population total estimator by Horvitz-Thompson, the ratio estimator and the estimator proposed by (Dorfman, 1992) using the unconditional bias, conditional bias and the Mean Squared Error obtained from a simulated data using some selected theoretical data variables.

CHAPTER TWO

LITERATURE REVIEW

2.1 Introduction

In order to have estimators of finite population parameters which are of high precision in sample surveys, statisticians employ the use of auxiliary information. This chapter reviews the relevant literature on the studies that have been done so far in relation to nonparametric estimations of finite population parameters.

2.2 Nonparametric Regression

The idea of data exploration using nonparametric regression methods has history of introduction. A regression model summarizes the relationship between two variables X and Y by quantifying the contributions of the explanatory variable X to the survey variable Y. This relationship is modelled as

$$Y_i = m(X_i) + \varepsilon_i \tag{2.1}$$

$\boldsymbol{Y_i}$ Variable of interest

$\boldsymbol{m}$ An unknown function to be determined using the sample data

$\boldsymbol{X_i}$ Auxiliary variable

$\boldsymbol{\varepsilon_i}$ Error term assumed to be $N(0, \sigma^2)$

for n data points.

There are four main approaches used in the estimation of finite population totals in sample surveys; model-based approach, design-based approach, model-assisted approach and design-assisted approach.

2.3 Review of the Approaches.

2.3.1 Design-Based Approach

In this approach, the observed values of the variable of interest say Y given as $y_{1,}y_{2,} \ldots >, y_n$ of the target population are viewed as unknown but fixed constants. This implies that the sample measurements of the samples drawn from the finite population are used in the estimation of the population parameter of interest. That is, the sample selection probabilities are used to provide the basis for inference. This approach is also referred to as the Classical/Randomization theory. Randomization theory in a manner, gives a nonparametric approach to inference where assumptions about the distribution of the random variables are not made (Lohr, 2021). According to (Horvitz and Thompson, 1952), the design-based expansion estimator of the population total is thus given as,

$$\hat{T}_{HT} = \sum_{i\epsilon s} \frac{1}{\pi_i} y_i \tag{2.2}$$

Where π_i is the inclusion probability given as, $\pi_i = \frac{n}{N}$

Statisticians rely on this approach due to its ability of removing biases during sample selection and usage in cases where not much is known about the population. Here researchers focus on having design-unbiased methods of estimation and apply less effort on the nature of the population itself. This approach shows the way the sample is selected and thus the distribution is well known because it is established on the population by the designer.

However, the major drawback of this approach is the assumption that all the samples in the population are selected which is impossible due to the problems associated with the selection of samples (Godambe, 1955). Therefore, optimality and robustness cannot be achieved simultaneously under this approach.

2.3.2 Model-Based Approach

In the model-based approach, the distribution is a structure existing in the population itself and is unexplored but capable of being modelled. In this forecast approach, the expectations are over all possible realizations of a linear regression stochastic model linking the study variable Y with a set of auxiliary variables X. In model-based approach, the actual values for the finite population $y_1, y_2, \ldots, y_n$ are treated as the realizations of the random variables $Y_1, Y_2, .., Y_N$. In the presence of auxiliary information, statisticians assume a working superpopulation model to describe the relationship between the variable of interest and the set of auxiliary variables. We assume that Y is a function of X, hence we have the model

$$Y_i = m(X_i) + \varepsilon_i$$

for i=1, 2, …., N where $m(X)$ is a smooth function and e_i are assumed to be normally identically and independently distributed error component terms with mean zero and a finite variance. Here, the estimator of the population total is given as

$$\hat{T} = \sum_{i=1}^{N} Y_i \tag{2.3}$$

which can be written as

$$\hat{T} = \sum_{i\epsilon s} Y_i + \sum_{i\epsilon p-s} Y_i \tag{2.4}$$

where the first part represents the sampled proportion and the second part represents the non-sampled proportion. The prediction of the non-sample part relies on the information obtained from the sample part. According to (Dorfman, 1992), the non-sampled part is estimated nonparametrically to give the model-based estimator of finite population total as,

$$\hat{T} = \sum_{i\epsilon s}^{N} Y_i + \sum_{i\notin s}^{N} \hat{m}(X_i) \tag{2.5}$$

Where $\hat{m}(X_i) = \sum_i w_i(x) Y_i$.

2.3.3 Model-Assisted Approach

This is a well-known approach that includes auxiliary information into the design-based estimation of finite population total. It works under the assumption of the existence of a superpopulation model between the auxiliary variable and the variable of interest for the population to be sampled.

Here, the model is used in improving the efficiency of the estimators and the estimators remain design consistent even when the model is incorrect. That is, inferences basically are design-based while the model serves as a way of helping choose between the randomization-based methods (Lang'at et al 2007). Due to their potential of improving the precision of the survey estimators in the availability of appropriate auxiliary information, these models are required to be linear or have a known parametric shape (Chaudhuri and Stenger, 2005). The model-assisted estimator of the finite population is given as,

$$\hat{T} = \sum_{i\epsilon U} \hat{y}_i + \sum_{i\epsilon s}(y_i - \hat{y}_i)\pi^{-1} \tag{2.6}$$

Where the first part represents the predicted values over the population and the second part represents the sample mean of residuals or the estimate of bias over the sample units and π is the inclusion probability.

In this study, we considered a model-based approach because it gives out consistent results compared to the other approaches. Additionally, model-based approach is the best option since we need results that are both optimal and robust.

In the following subsection we review different studies that have been carried out using these approaches in nonparametric regression.

2.4 Review of Selected Nonparametric Estimators

The idea of nonparametric estimation methods was first introduced by (Nadaraya, 1964) and (Watson, 1964). It was introduced in the estimation of a regression curve using the model

Y= m(x_i) + $\delta(x_i)e$ where m(x) is the smoothing function, e is a random error component with a mean of zero and a constant finite variance. The objective of their paper was in estimating the smoothing function m(x). The N-W estimator of the smooth function is given by

$$\hat{m}(x) = \sum_{ies} w_i(x) y_i \tag{2.7}$$

(Dorfman, 1992), used a nonparametric regression estimator for finite population totals based on a sample drawn from a population using a simple kernel estimator. He used the N-W weights in obtaining the Nadaraya-Watson estimator. Simulation studies carried out found that the estimator is more efficient than the design-based estimators and if the bandwidth is larger, the density function becomes broader and flatter, the more equal are the weights and the smoother the estimated function. (Orwa et al, 2010) proposed a nonparametric regression approach of a finite population total in model-based framework in the case of a stratified sampling. The estimator was based on the modified N-W kernel estimator and it led to relatively small error.

(Breidt and Opsomer, 2000) considered nonparametric method on with a design-based approach. They used the local polynomial regression estimator for unknown regression function $m(X)$ which is used as a generalization of the ordinary generalized regression estimator. Their assumption was that m(x) is a smooth function of x and used the model to obtain design-unbiased and consistent estimators of the finite population total. (Ombui, 2008) used local polynomial in estimating finite population parameters.

(Kim et al, 2009) adapted the (Breidt and Opsomer, 2000) local polynomial nonparametric regression estimation to two-stage cluster sampling. A probability sample of clusters is drawn from the population of clusters according to a fixed size design and then subsamples of every sampled cluster were obtained. They assumed that the inclusion probabilities were strictly positive and the variance and independence of the two-stage design. The two-stage design is frequently used because an adequate frame of elements is not available or would be prohibitively

expensive to construct, but a listing of clusters is available. The results from the simulation study found that nonparametric methodology compares favorably with Horvitz-Thompson and classical survey estimates. (Syengo, 2018) considered local polynomial regression under stratified random sampling in the estimation of finite population totals. The population of interest is divided into strata, a simple random sample is selected without replacement from a stratum and the size of the sample should be sufficiently large. The estimates of the study were found to be asymptotically unbiased and consistent.

(Breidt and Opsomer,2005) considered a nonparametric design-based regression estimator based on penalized splines. He suggests that they can be used to improve the efficiency of estimators in situations where linear models are not appropriate and are also easy to be incorporated into more complicated models like the additive semiparametric models. (Zheng and Little, 2003) in their paper used penalized splines under model-based approach to estimate finite population totals.

2.4.1 Reflection of Data Method

Reflection method is a statistical technique that was introduced early on by scholars. The basic idea in this method is to reflect the data points at the origin and work with them. It is used in the reduction of bias problems encountered at the boundaries. If certain conditions were not fulfilled, reflection does not yield satisfying results always as it contains a bias of order h.

(Schuster, 1985) used reflection of data method in density estimation. This method was used as a technique of reducing the boundary bias. To add on, (Alberts & Karunamuni, 2006) reviewed the use of data reflection method in their study on the methods of boundary correction in kernel density estimation.

(Lang'at, 2017) studied robust estimation of finite population total in nonparametric regression incorporating data reflection method. The estimator was under model-based framework. He

found out that a good number of kernels have the suitable features for application. The estimator found reduced the boundary bias significantly thus it was superior than all other apart from where the ratio estimator dominated in linear model. In their study (Lang'at et al, 2019), explored nonparametric estimation of finite population total under model-based framework. They used kernel smoother in the construction of the estimator. However, this estimator suffers boundary problems which they catered for by modifying it by the use of reflection technique. This estimator was found to be favorably okay compared to other estimators.

2.4.2 Transformation Method

This technique was introduced in a study by (Wand et al, 1991). Here, one can take a one-to-one function which is continuous and then a regular kernel estimator is used with the transformed data.

(Karunamuni and alberts, 2005) applied transformation method in their study and the estimator was found to be locally adaptive and non-negative if the kernel function was non-negative. Their approach had a high potential of producing better estimators. Also, (Karunamuni and Alberts, 2006) applied a locally adaptive transformation method of boundary correction in kernel density estimation. The method was computationally easy and convenient. They found out that the amount of transformation was dependent upon the estimation point. Their estimator depends on the density function applied. (Bii et al, 2020) used a transformation of data method in estimating finite population mean. They concluded that their proposed estimator provided a better estimation of the mean of a finite population compared to other estimators. (Bii et al, 2019) studied boundary bias correction using weighting method in two stage cluster sampling. A a modified transformation of data method was used in the estimation. Their estimator was found to produce estimates that were closer to the true population values being estimated.

In this study, we applied the hybrid of the two methods in the estimation of finite population total and its performance was compared against the other existing estimators such as the ratio estimator which is the Best Linear Unbiased Predictor (Cochran, 1977) given as,

$$\hat{T}_R = \hat{B} \sum_{i=1}^{N} X_i \tag{2.8}$$

Where $\hat{B} = \frac{\sum_{i=1}^{n} y_i}{\sum_{i=1}^{n} x_i}$, represents the estimator for the population parameter, $\sum_{i=1}^{n} y_i$ represents the sample total of the study variable and $\sum_{i=1}^{n} x_i$ represents the estimator of the auxiliary variable assumed to be known for the population. Population total for this auxiliary variable is given as $\sum_{i=1}^{N} X_i$.

The design-based Horvits-Thompson estimator proposed by (Thompson, 1952) given by,

$$\hat{T}_{HT} = \sum_{i \epsilon s} \pi_i^{-1} y_i \tag{2.9}$$

Where π_i represents the inclusion probability.

and the model-based nonparametric estimator proposed by (Dorfman, 1992) given by,

$$\hat{T}_{np} = \sum_{i=1}^{n} y_i + \sum_{i=n+1}^{N} \hat{m}_{NW}(x_i) \tag{2.10}$$

Where the first part represents the sample proportion and the second part represents the Nadaraya-Watson estimator.

2.5 Research Gap

Several studies have been carried out using various nonparametric regression methods to estimate the finite population totals. However, these regression methods have applied kernel smoothers to estimate the regression functions and the smoothers are known to suffer the problem of boundary bias and different researchers have worked out to come up with estimators to address the problem. Different methods of reducing boundary problem including the reflection

have been used to come up with estimators of finite population total but a hybrid method has not been covered.

This study focuses on the removal of the boundary bias significantly by the use of a hybrid of data transformation and data reflection technique to the nonparametric estimation for finite population total since it’s an area that has not been explored.

CHAPTER THREE

RESEARCH METHODOLOGY

3.1. Introduction

The general objective of the study was to propose a nonparametric estimator for finite population total using a hybrid of data reflection and transformation of data under model-based framework. This chapter introduces some of the basic terminologies used in statistics and then we focused on the methods of estimating finite population totals using a hybrid of data-reflection and transformation technique nonparametrically.

3.2. Finite and Infinite Population

Basically, population can be categorized into two types, which is the finite and infinite population. In finite population, it is possible to enumerate the total number of individuals or units it contains. On the other hand, in infinite population it is not possible to enumerate the total number of units or individuals contained in the population.

3.3. Sources of Data

In this study, the data for testing the efficiency of the proposed estimator was obtained through simulation in R statistical package.

Let $x_1, x_2, \dots, x_n$ be random samples and $y_1, y_2, \dots, y_n$be the study variables. The random variables were used in generation of the artificial data set. We employed different models in the simulation of the data which include, the linear model, quadratic model and the exponential theoretical models. Simulation is a method of getting computer generated data through experiments by random sampling. However, the strength of simulation is its ability in understanding the behaviors of statistical methods.

3.4. Review of Estimation Methods

Let $X_1, X_2, \dots, X_N$ be independent and identically distributed random variables with continuous distribution function. Further, let there be a sample of size n and a kernel function k which is symmetric around the origin. Therefore, the standard kernel density estimator is given as;

$$\widehat{m}(X_i) = \frac{1}{nh}\sum_{i=1}^{k} k\left(\frac{x-X_i}{h}\right) \tag{3.1}$$

Where h is the bandwidth and k is a non-negative integrable smoothing kernel.

3.4.1 Data Reflection Technique

Data reflection is one of the methods used in boundary bias reduction. In our study, we proposed transformation in corporation with reflection technique to address the problem of boundary biases.

Let $\{(X_1, Y_1), (X_2, Y_2), \dots, (X_n, Y_n)\}$ be the set of n observations from the sample. Under reflection of all the points in the boundary, the data increases in number to give the new set of data of the form, $\{(X_1, Y_1), (-X_1, -Y_1), (X_2, Y_2), (-X_2, -Y_2), \dots, (X_n, Y_n), (-X_n, -Y_n)$. Therefore, the kernel estimate obtained from this estimate is of size 2n. The standard kernel estimator for this method is written as

$$\widehat{m}_R(x) = \frac{1}{2nh}\sum_{j=1}^{2n} k\left(\frac{x-X_i}{h}\right), \quad x \in \mathbb{R} \tag{3.2}$$

This can be written as,

$$\widehat{m}_R(x) = \frac{1}{nh}\sum_{i=1}^{n} \left\{k\left(\frac{x-X_i}{h}\right) + k\left(\frac{x+X_i}{h}\right)\right\} x \geq 0 \tag{3.3}$$

3.4.2 Transformation of Data Method

We assumed m has unknown probability density function with support $[0, \infty)$ Consider a random sample of $x_1, x_2, \dots, x_n$ from m used in estimating m. The idea behind transformation is based on transforming the original data $X_1, X_2, \dots, X_N$ through a function g to obtain a transformed data

given as $g(X_1), g(X_2), \dots, g(X_N)$. Here, g is a positive, continuous and monotonically increasing function. From the standard kernel estimator, the transformed kernel density estimator is of the form,

$$\widehat{m}_T(x) = \frac{1}{nh}\sum_{i=1}^{n} k\left(\frac{x-g(X_i)}{h}\right) \tag{3.4}$$

Where h is the bandwidth and k is a symmetric positive kernel function.

3.4.3 The Proposed Estimator

Following the standard form of the kernel estimator, a nonparametric estimator of finite population is proposed. The proposed hybrid method was obtained by combining data transformation and data reflection techniques to come up with the superior method of estimation of the kernel estimator. Given the two formulas, $\widehat{m}_R(x) = \frac{1}{nh}\sum_{i=1}^{n}\left\{k\left(\frac{x-X_i}{h}\right) + k(\frac{x+X_i}{h})\right\}$ and $\widehat{m}_T(x) = \frac{1}{nh}\sum_{i=1}^{n} k\left(\frac{x-g(X_i)}{h}\right)$, we attain the proposed estimator by combining the two to have,

$$\widehat{m}_{RT}(x) = \frac{1}{nh}\sum_{i=1}^{n}\left\{k\left(\frac{x-g_1(X_i)}{h}\right) + k\left(\frac{x+g_2(X_i)}{h}\right)\right\} \tag{3.5}$$

Where h is the bandwidth, k is the kernel function, $g_1 and\ g_2$ are transformations that were determined. For convenience it was assumed that $g_1 = g_2$ for this study.

3.5. Estimation of Finite Population Totals

Here the procedure for finite population totals estimation is presented. We assume that we have $U_1, U_2, \dots, U_N$ sampling units corresponding to the survey measurements $Y_1, Y_2, \dots, Y_N$ so that the population total denoted by T is defined as

$$\text{T}=\sum_{i=1}^{N} Y_i \tag{3.6}$$

We use the model

$$Y_i = m(X_i) + e_i \tag{3.7}$$

as the estimator of the equation (3.6) since it's the simplest equation that describes the relationship between the auxiliary variable and the study variable. Suppose the auxiliary information $X_1, X_2, .., X_N$ is available for these Y_i's and are to be considered in the estimation process. Then the equation takes the prediction form

$$T=\sum_{i\epsilon s} Y_i + \sum_{i\notin s} Y_i \tag{3.8}$$

Where the first part presents the truly sampled proportion and the second part presents the proportion to be estimated using the auxiliary information available. Several estimation methods have been used to address the problem. An estimator of the form

$$\hat{T} = \sum_{i\epsilon s} Y_i + \sum_{i\notin s} \hat{Y}_i \tag{3.9}$$

is looked at in this study. To the unobserved part of the equation, we use equation (3.7) and the equation takes the form

$$\hat{T} = \sum_{i\epsilon s} Y_i + \sum_{i\notin s} m(x) \tag{3.10}$$

Now the task reduces to estimating the second part of the equation. To tackle this, we used a hybrid formula incorporating data transformation and reflection techniques which leads to the proposed estimator given as,

$$\hat{T}_{RT} = \sum_{i\epsilon s} Y_i + \sum_{i\notin s} \left\{ \frac{1}{nh} \sum_{i=1}^{n} \left[k\left(\frac{x-g_1(X_i)}{h}\right) + k\left(\frac{x+g_2(X_i)}{h}\right) \right] \right\} \tag{3.11}$$

In the estimation process, the data was first transformed using the quadratic function given as $g(x) = x^2 + 2x + 2$. Where $g(x)$ is non negative continuous and monotonically increasing quadratic function. Secondly, the transformed data was then reflected and the analysis done.

The efficiency and unbiasedness of the estimator were tested both analytically and empirically.

3.6. The Big "Oh" and Little "oh" Notations

Given two functions $f(x)$ and $g(x)$ of a real variable x considered as $x \to \infty$, the expression relating the two functions for sufficiently large x is given as;

$$f(x) = O\big(g(x)\big) \tag{3.12}$$

This implies that, there exists a constant a and x_o such that

$$|f(x)| \leq a|g(x)| \quad \text{for } x \geq x_o \tag{3.13}$$

If the equality holds, then $f(x)$ is said to be of $O\big(g(x)\big)$.

Consequently,

$$f(x) = o\big(g(x)\big) \quad \text{as } x \to \infty \tag{3.14}$$

This implies that $g(x) \neq 0$ for sufficiently large x and $\lim_{n\to\infty} \frac{f(x)}{g(x)} = 0$

If the relation holds, then $f(x)$ is said to be of smaller order than $g(x)$. This is as reviewed by (Cormen et al, 2022) in introduction to algorithms.

3.7. Properties of the Proposed Estimator

In the estimation process, the properties of the proposed estimator are desirable. According to (Tsybakov, 2008), efficiency and bias are key properties of estimators that statisticians should be eager to investigate as they measure the amount of accuracy and precision of estimators. But efficiency is a function of variance, thus we end up studying variance and bias. The Mean Squared Error (MSE) is used as the basic measure of accuracy that accounts for both the bias and variance of an estimator at an arbitrary fixed point.

3.8.Bias of the Proposed Estimator

Theorem 1

The assumptions used by (cowling and hall, 1996) are used in deriving the bias and variance of the proposed estimator. Assume the transformation g_i i=1,2 is non-negative continuous and monotonically increasing functions defined on [0,∞). Further, assume that g_i^{-1} exists $g_i(0)$=2, $g_i'(0)$=2 and that g'' and g''' and are continuous on [0,∞) where $g_i^{(j)}$ denotes the jth derivative

of g_i with $g_i^{(0)}$=g_i and g_i^{-1} denoting the inverse function of g_i, i=1,2. Suppose that m^j is the j^{th} derivative of m and that it exists and is continuous on [0,∞), j=0,1,2 with $m^{(0)}$=m. Furthermore, let x =ch where $0 \leq c \leq 1$. Assume the kernel function k is non-negative symmetric function with support [-1,1] such that it satisfies

$$\int k(t)dt = 1, \qquad \int tk(t)dt = 0, \quad and \;\; 0 < \int t^2 k(t)dt < \infty \tag{3.15}$$

that is, K is a kernel of order 2.

The bias of the proposed estimator is given as;

$$E[\hat{m}_{RT}(x_i)] - m(x)$$

Proposition 3.1

$$\text{Bias}(\hat{T}_{RT}) = \frac{N-n}{n}\Big\{2h\Big[2m'(0)\int_c^1 (t-c)K(t)dt - g_1''(0)m(0)\int_c^1 (t-c)K(t)dt \; - g_2''(0)m(0)\Big(c + \int_c^1 (t-c)K(t)dt\Big)\Big] + \frac{2h^2}{2}\Big[-c^2 m''(0) + m''(0)\int_{-1}^1 (t-c)^2 K(t)dt - [g_1'''(0)m(0) + 3g_1''(0)\{2m'(0) - g_1''(0)m(0)\}]\int_c^1 (t-c)^2 K(t)dt - [g_2'''(0)m(0) + 3g_2''(0)\{2m'(0) - g_2''(0)m(0)\}]\int_{-1}^c (t-c)^2 K(t)dt\Big]\Big\} + o(h^2) \tag{3.16}$$

Proof

Under the model-based approach the bias of an estimator is given as,

$$\text{Bias}(\hat{T}_{RT}) = E(\hat{T} - T)$$

$$= E([\sum_{i=1}^n y_i + \sum_{i=n+1}^N \hat{m}_{RT}(x_i)] - [\sum_{i=1}^n y_i + \sum_{i=n+1}^N y_i])$$

$$= E(\sum_{i=n+1}^N \hat{m}_{RT}(x_i) - \sum_{i=n+1}^N y_i)$$

$$\text{Bias}(\hat{T}_{RT}) = E(\sum_{i=n+1}^N \hat{m}_{RT}(x_i) - \sum_{i=n+1}^N m(x)) \tag{3.17}$$

The proposed estimator is given as,

$$\widehat{m}_{RT}(x) = \frac{1}{nh}\sum_{i=1}^{n}\left\{K\left(\frac{x-g_1(X_i)}{h}\right) + K\left(\frac{x+g_2(X_i)}{h}\right)\right\}$$

$$\mathrm{E}(\widehat{m}_{RT}(x_i)) = \frac{1}{nh}E\left\{\sum_{i=1}^{n}K\left(\frac{x-g_1(X_i)}{h}\right) + \left(\frac{x+g_2(X_i)}{h}\right)\right\}$$

$$\sum_{i=n+1}^{N}[(\widehat{m}_{RT}(x_i))] = \frac{1}{nh}\sum_{i=n+1}^{N}\left\{E\sum_{i=1}^{n}\left[K\left(\frac{x-g_1(X_i)}{h}\right) + K\left(\frac{x+g_2(X_i)}{h}\right)\right]\right\} \quad (3.18)$$

Analyzing the first part of the equation 3.18

$$= \frac{1}{nh}\sum_{i=n+1}^{N}\left\{\sum_{i=1}^{n}E\left[K\left(\frac{x-g_1(X_i)}{h}\right)\right]\right\}$$

$$= \frac{N-n}{nh}\int_0^{\infty}K\left(\frac{x-g_1(X_i)}{h}\right)m(X_i)dX$$

Using change of variable technique, we have,

Let $t = \left(\frac{x-g_1(X_i}{h}\right)$ and given that x=ch

$$g_1(X_i) = ch - th$$

$$X_i = g_1^{-1}(c-t)h$$

Thus, the equation becomes,

$$\sum_{n+1}^{N}[\widehat{m}_{RT}(x_i)] = \frac{N-n}{n}\int_{-1}^{c}K(t)\frac{m\left(g_1^{-1}(c-t)h\right)}{g_1'\left(g_1^{-1}(c-t)h\right)}\mathrm{dt} \quad (3.19)$$

Using Taylor series expansion of order 2 at t=c

$$= \frac{N-n}{n} \int_{-1}^{c} \left\{ \frac{m(g_1^{-1}(0))}{g_1'(g_1^{-1}(0))} \right.$$

$$- (t-c)h \frac{g_1'(g_1^{-1}(0))m'(g_1^{-1}(0)) - g_1''(g_1^{-1}(0))m(g_1^{-1}(0))}{[g_1'(g_1^{-1}(0))]^3}$$

$$+ \frac{h^2}{2}(t-c)^2 \left[\frac{g_1'(g_1^{-1}(0))m''(g_1^{-1}(0)) - g_1'''(g_1^{-1}(0))m(g_1^{-1}(0))}{[g_1'(g_1^{-1}(0))]^4} \right.$$

$$\left. \left. - \frac{3g_1''(g_1^{-1}(0))\{g_1'(g_1^{-1}(0))m'(g_1^{-1}(0)) - g_1''(g_1^{-1}(0))m(g_1^{-1}(0))\}}{[g_1'(g_1^{-1}(0))]^5} \right] \right\} dt + o(h^2)$$

(3.20)

Using the assumptions $g^{-1}(0) = 0 \; and \; g'(0) = 2$ the equation reduces to

$$= \tfrac{N-n}{n} \Big\{ m(0) \textstyle\int_{-1}^{c} K(t)dt - 2h \int_{-1}^{c} (t-c)K(t)dt[m'(0) - g_1''(0)m(0)] + \tfrac{2h^2}{2} \int_{-1}^{c} (t-$$

$$c)^2 K(t)dt\{m''(0) - g_1'''(0)m(0) - 3g_1''(0)[2m'(0) - g_1''(0)m(0)]\} \Big\} + o(h^2)$$

(3.21)

For the second part of the equation we have,

$$= \tfrac{1}{nh} \textstyle\sum_{i=n+1}^{N} \left\{ \sum_{i=1}^{n} E \left[K \left(\frac{x+g_2(X_i)}{h} \right) \right] \right\}$$

$$= \tfrac{N-n}{nh} \textstyle\int_0^{\infty} K \left(\frac{x+g_2(X_i)}{h} \right) m(X_i) dX \qquad (3.22)$$

Using change of variables technique, we have,

$$= \frac{N-n}{n} \int_c^1 K(t) \frac{m(g_2^{-1}(t-c)h)}{g_2'(g_2^{-1}(t-c)h)} \mathrm{dt} \qquad (3.23)$$

Under Taylor series expansion of order 2 at t=c the equation becomes,

$$= \frac{N-n}{n}\int_c^1 \left\{\frac{m(g_2^{-1}(0))}{g_2'(g_2^{-1}(0))} + (t-c)h\frac{g_2'\left(g_2^{-1}(0)\right)m'(g_2^{-1}(0)) - g_2''\left(g_2^{-1}(0)\right)m\left(g_2^{-1}(0)\right)}{\left[g_2'\left(g_2^{-1}(0)\right)\right]^3} + \frac{h^2}{2}(t-$$

$$c)^2\left[\frac{g_2'(g_2^{-1}(0))m''(g_2^{-1}(0)) - g_2''(g_2^{-1}(0))m(g_2^{-1}(0))}{[g_2'(g_2^{-1}(0))]^4} -\right.$$

$$\left.\left.\frac{3g_2''\left(g_2^{-1}(0)\right)\{g_2'\left(g_2^{-1}(0)\right)m'(g_2^{-1}(0)) - g_2''\left(g_2^{-1}(0)\right)m\left(g_2^{-1}(0)\right)\}}{[g_2'(g_2^{-1}(0))]^5}\right]\right\}dt + o(h^2) \qquad (3.24)$$

Since $m''(0)$ exists and is continuous near 0, for $x = ch$ we have,

$$m(0) = m(x_i) - chm'(x_i) + \frac{(ch)^2}{2}m''(x_i) + o(h^2)$$

$$m'(x) = m'(0) + chm''(0) + o(h)$$

$$m''(x) = m''(0) + o(1)$$

$$E(\widehat{m}_{RT}(x_i)) = \frac{N-n}{n}\left\{m(0) + 2h\left[\int_c^1 (t-c)K(t)dt\{m'(0) - g_1''(0)m(0)\}\right] -\right.$$

$$2h\left[\int_{-1}^{c}(t-c)K(t)dt\{m'(0) - g_2''(0)m(0)\}\right] + \frac{2h^2}{2}\left\{\int_c^1 (t-c)^2K(t)dt[m''(0) -\right.$$

$$g_1'''(0)m(0) - 3g_1''(0)\{2m'(0) - g_1''(0)m(0)\}]\Big\} + \frac{2h^2}{2}\left\{\int_{-1}^{c}(t-c)^2K(t)dt[m''(0) -\right.$$

$$\left.\left.g_2'''(0)m(0) - 3g_2''(0)\{2m'(0) - g_2''(0)m(0)\}]\right\}\right\} + o(h^2)$$

$$= m(x) + 2h\left\{2m'(0)\int_c^1 (t-c)K(t)dt - g_1''(0)m(0)\int_c^1 (t-c)K(t)dt -\right.$$

$$g_2''(0)m(0)\left(c + \int_c^1 (t-c)K(t)dt\right)\Big\} + \frac{2h^2}{2}\Big\{-c^2m''(0) + m''(0)\int_{-1}^{1}(t-$$

$$c)^2K(t)dt - [g_1'''(0)m(0) + 3g_1''(0)\{2m'(0) - g_1''(0)m(0)\}]\int_c^1 (t-c)^2K(t)dt -$$

$$\left.[g_2'''(0)m(0) + 3g_2''(0)\{2m'(0) - g_2''(0)m(0)\}]\int_{-1}^{c}(t-c)^2K(t)dt\right\} + o(h^2)$$

(3.25)

Thus, the bias is given as,

$$E[\widehat{m}_{RT}(x_i)] - m(x)$$

$$= \frac{N-n}{n}\left\{2h\left[2m'(0)\int_c^1 (t-c)K(t)dt - g_1''(0)m(0)\int_c^1 (t-c)K(t)dt \; - \right.\right.$$

$$g_2''(0)m(0)\left(c + \int_c^1 (t-c)K(t)dt\right)\Big] + \frac{2h^2}{2}\Big[-c^2 m''(0) + m''(0)\int_{-1}^1 (t-c)^2 K(t)dt -$$

$$[g_1'''(0)m(0) + 3g_1''(0)\{2m'(0) - g_1''(0)m(0)\}]\int_c^1 (t-c)^2 K(t)dt - [g_2'''(0)m(0) +$$

$$\left.\left. 3g_2''(0)\{2m'(0) - g_2''(0)m(0)\}]\int_{-1}^c (t-c)^2 K(t)dt\Big]\right\} + o(h^2)$$

(3.26)

As $n \to \infty \; and \; h \to 0$ the bias of the estimator tends to zero.

3.9.Variance of the Proposed Estimator.

The variance of the proposed estimator is given by proposition 3.9

Proposition 3.2

$$Var(\widehat{m}_{RT}(x)) = \frac{(N-n)^2}{nh}\left\{\int_{-1}^1 K(t)^2 dt + 2\int_{-1}^c K(t)K(-2+(3c-2t))dt\right\} + o\left(\frac{1}{nh}\right)$$

(3.27)

Proof

$$var(T) = E[T]^2 - [E(T)]^2$$

$$var[\textstyle\sum_{i=n+1}^N (\widehat{m}_{RT})] = \frac{(N-n)^2}{nh^2}\left\{var\left[K\left(\frac{x-g_1(X_i)}{h}\right) + K\left(\frac{x+g_2(X_i)}{h}\right)\right]\right\}$$

$$= \frac{(N-n)^2}{nh^2}\left\{E\left[K\left(\frac{x-g_1(X_i)}{h}\right) + K\left(\frac{x+g_2(X_i)}{h}\right)\right]^2 - \left[E\left(K\left(\frac{x-g_1(X_i)}{h}\right) + \right.\right.\right.$$

$$\left.\left.\left. K\left(\frac{x+g_2(X_I)}{h}\right)\right)\right]^2\right\}$$

We let,

$$A = \frac{(N-n)^2}{nh^2}\left\{E\left[K\left(\frac{x-g_1(X_i)}{h}\right) + K\left(\frac{x+g_2(X_i)}{h}\right)\right]^2\right\}$$

$$= \frac{(N-n)^2}{nh^2}\left\{E\left[K\left(\frac{x-g_1(X_i)}{h}\right)\right]^2 + E\left[K\left(\frac{x+g_2(X_i)}{h}\right)\right]^2 + 2E\left[K\left(\frac{x-g_1(X_i)}{h}\right)K\left(\frac{x+g_2(X_i)}{h}\right)\right]\right\}$$

$$= \frac{(N-n)^2}{nh^2}\left\{\int_0^\infty K\left(\frac{x-g_1(X_i)}{h}\right)^2 m(X)dX + \int_0^\infty K\left(\frac{x+g_2(X_i)}{h}\right)^2 m(X)dX + \right.$$

$$\left. 2\int_0^\infty K\left(\frac{x-g_1(X_i)}{h}\right)K\left(\frac{x+g_2(X_i)}{h}\right)m(X)dX\right\} \quad (3.28)$$

Using the change of variable technique, by letting $X_i = u$, we have,

$$= \frac{(N-n)^2}{nh^2}\left\{\int_0^\infty K\left(\frac{x-g_1(u)}{h}\right)^2 m(u)du + \int_0^\infty K\left(\frac{x+g_2(u)}{h}\right)^2 m(u)du + \right.$$

$$\left. 2\int_0^\infty K\left(\frac{x-g_1(u)}{h}\right)K\left(\frac{x+g_2(u)}{h}\right)m(u)du\right\} \quad (3.29)$$

$$= A_1 + A_2$$

Computing A_1 ,

Let $t = \frac{x-g_1(u)}{h}$ and given $x = ch$, we have

$$ht = ch - g_1(u)$$

$$u = g_1^{-1}(c-t)h$$

We have,

$$= \frac{(N-n)^2}{nh^2}\left[h\int_{-1}^{c} K^2(t)\frac{m\left(g_1^{-1}((c-t)h\right)}{g_1'\left(g_1^{-1}((c-t)h)\right)}dt + h\int_c^1 K^2(t)\frac{m\left(g_2^{-1}((t-c)h\right)}{g_2'\left(g_2^{-1}((t-c)h)\right)}dt\right] \quad (3.30)$$

$$= \frac{(N-n)^2 m(0)}{nh}\int_{-1}^{1} K(t)^2 dt + o\left(\frac{1}{nh}\right) \quad (3.31)$$

By the continuity property of $g_1'' \; and \; g_2''$ and by Taylor expansion of order two on g_1 and g_2, we have,

$$g_1\big((c-t)h\big) = g_1(0) + (t-c)(-h)g_1'(0) + O(h^2)$$

$$= 2 + 2(c-t)h + O(h^2) \tag{3.32}$$

And,

$$g_2\big((c-t)h\big) = g_2(0) + (t-c)(-h)g_2'(0) + O(h^2)$$

$$= 2 + 2(c-t)h + O(h^2) \tag{3.33}$$

Since $g_i(0) = 2$ and $g_i'(0) = 2$, i=1,2 using the two equations above and by the change of variables

$$t = \frac{x - g_1(X_i)}{h}$$

$$X_i = g_1^{-1}(x - ht)$$

$$A_2 = \frac{2(N-n)^2}{nh^2}\left\{\int K\left(\frac{x+g_1(X_i)}{h}\right)K\left(\frac{x-g_2(X_i)}{h}\right)m(X_i)dX\right\}$$

$$= \frac{2(N-n)^2}{nh}\left\{\int_{-1}^{c} K(t)K\left(\frac{x-g_2\left(g_1^{-1}(ht-x)\right)}{h}\right)m\big(g_1^{-1}(ht-x)\big)dt\right\}$$

$$= \frac{2(N-n)^2}{nh}\left\{\int_{-1}^{c} K(t)K\left(\frac{x-(2+2(t-c)h+O(h^2))}{h}\right)m(g_1^{-1}(ht-x))dt\right\}$$

$$= \frac{2(N-n)^2}{nh}\int_{-1}^{c} K(t)K(-2 + \big(3c - 2t + O(h)\big))(m(0) + O(h))dt$$

$$= \frac{2(N-n)^2}{nh}\int_{-1}^{c} K(t)K(-2 + (3c - 2t))dt + o\left(\frac{1}{nh}\right) \tag{3.34}$$

$$B = \frac{1}{nh^2}\left\{E\left[K\left(\frac{x+g_1(x_i)}{h}\right) + K\left(\frac{x-g_2(x_i)}{h}\right)\right]\right\}^2$$

$$= o\left(\frac{1}{nh}\right)$$

This follows from equations 3.16 and 3.19

Therefore, by combining the equations above (3.22 to 3.28), we have

$$Var(\hat{m}_{RT}(x)) = \frac{(N-n)^2}{nh}\left\{\int_{-1}^{1} K(t)^2 dt + 2\int_{-1}^{c} K(t)K(-2+(3c-2t))dt\right\} + o\left(\frac{1}{nh}\right)$$

(3.35)

The variance of $\hat{m}(x)$ decreases in nh as $n \to \infty$ and the bandwidth h$\to$ 0. Therefore, this implies that as variance decreases with an increase in sample size.

3.10. Mean Squared Error of the Proposed Estimator

The mean squared error brings together the variance of the estimator and the square of the bias term of the estimator

$$\text{MSE}(\hat{T}) = Var\left(\hat{T}\right) + (Bias)^2$$

$$= \frac{(N-n)^2}{nh}\left\{\int_{-1}^{1} K(t)^2 dt + 2\int_{-1}^{c} K(t)K(-2+(3c-2t))dt\right\} +$$

$$\left[\frac{N-n}{n}\left\{2h\left[2m'(0)\int_{c}^{1}(t-c)K(t)dt - g_1''(0)m(0)\int_{c}^{1}(t-c)K(t)dt\right.\right.\right. -$$

$$g_2''(0)m(0)\left(c+\int_{c}^{1}(t-c)K(t)dt\right)\Big] + \frac{2h^2}{2}\Big[-c^2m''(0) + m''(0)\int_{-1}^{1}(t-c)^2K(t)dt -$$

$$[g_1'''(0)m(0) + 3g_1''(0)\{2m'(0) - g_1''(0)m(0)\}]\int_{c}^{1}(t-c)^2K(t)dt - [g_2'''(0)m(0) +$$

$$\left.\left.\left.3g_2''(0)\{2m'(0) - g_2''(0)m(0)\}]\int_{-1}^{c}(t-c)^2K(t)dt\Big]\right\}\right]^2 + o\left(\frac{1}{nh}\right)$$

(3.36)

The mean squared error decreases in nh as n$\to \infty$ and $h \to 0$ which implies that the mean squared error decreases with an increase in sample size.

CHAPTER FOUR

RESEARCH RESULTS

4.1 Introduction

In this chapter, simulated data is used to test the developed theory in chapter three. The simulation of data was carried out using the R statistical package. The conditional and unconditional properties of the simulation variables are reviewed. Three mean functions are used in generating data sets that was used to carry out the estimation of the population total and the corresponding mean squared error. In the study, we used the mean functions for totals employed by (Breidt and Opsomer, 2000). They include linear, quadratic and exponential theoretical models. Afterwards, the analysis and comparison of the performance of the proposed estimator was done against the estimator proposed by (Dorfman, 1992), the ratio estimator and the Horvitz-Thompson estimator.

4.2 Properties of the Data Variables for Simulation

The auxiliary variables were generated as independent and identically distributed random variables on U(0,1). For the improvement on the precision of estimation, the auxiliary variables for each data set are collected and included in the estimators. This is due to the importance of the information contained in the auxiliary variable necessary for the estimation of population total. The data sets are artificial data obtained by simulating three theoretical models in R statistical package. The three theoretical data variables were adopted from (Breidt and Opsomer, 2000). The mean functions for getting the data sets are described below

The linear model was used to simulate the first data set. The model is given as

$$Y_i = 1 + 2(x_i - 0.5) + e_i \tag{4.1}$$

A uniform distribution is used to simulate the random variable X and takes the values that are equally likely to occur including the extremes from 0 to 1. (x_i, y_i) i= 1, 2, …., N are assumed to

be independent and identically distributed random variables with the error component being standard normal variable.

The second data set was obtained through simulation by the use of a quadratic model given as

$$Y_i = 1 + 2(x_i - 0.5)^2 + e_i \tag{4.2}$$

The random variable X here is also simulated by the use of a uniform distribution and takes the values that equally likely to occur including the extremes. (x_i, y_i) are assumed to be iid random variables and $e_i \sim N(0,1)$.

The third data set was simulated by the use of an exponential model given as

$$Y_i = \exp(-8x_i) + e_i \tag{4.3}$$

The random variable X is generated as independent and identically distributed U(0,1).

In all the three data variables, a population of size 1000 was simulated and samples of size 300 are selected from each population and the estimates of the population total and the mean squared error computed.

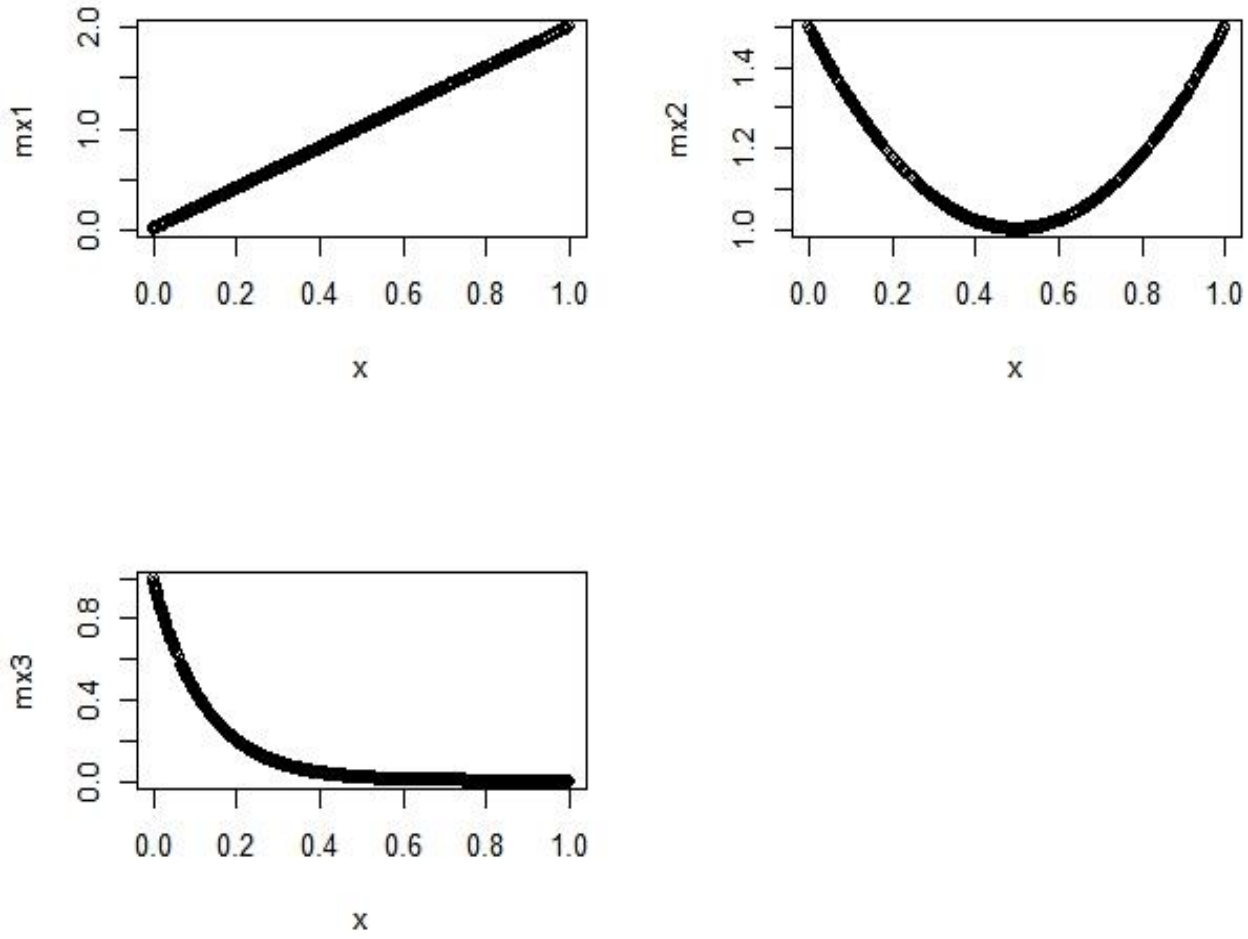

Figure 4.1a, 4.1b and 4.1c: Graphs showing the three data variable functions that is the linear model, the quadratic model and the exponential model respectively.

The figure 4.1a, 4.1b and 4.1c shows the properties of the three theoretical data variables used in data simulation. Figure 4.1a shows the linear property of the linear model, figure 4.1b represents the quadratic property of the equation and figure 4.1c shows the properties of the exponential function.

We chose to apply the three theoretical data variables in simulation due to their varied applications in real life. Linear functions are used in determining the relationship between the dependent and independent variables. For example, in economics linear functions are used to

analyze the relationship between the price, supply and demand of various commodities. Quadratic functions are used to develop profit and loss functions in economics. They are also used in physics to describe the trajectory followed by objects thrown at an angle. Exponential functions are used in modelling growth or decay. For example, it can be used to determine the population growth.

4.3 Unconditional Properties of the Estimator

In this section, the estimates of the bias and the mean squared error of the finite population total for the proposed estimator, the Dorfman estimator, the Horvitz-Thompson estimator and the Ratio estimator are recorded, analyzed and conclusions made. In the study, a population of size 1000 was simulated using statistical R i386 4.0.3 package and samples of size 300 generated using stratified random sampling. A comparison between the proposed estimator, the Dorfman estimator, the Horvitz-Thompson estimator and the ratio estimator was done. The biases of our estimator, the estimator proposed by Dorfman (1992), the Horvitz-Thompson estimator and the ratio estimator are computed as $(\hat{T}_{TR} - Y), (\hat{T}_{NW} - Y), (\hat{T}_{HT} - Y)\ and\ (\hat{T}_R - Y)$ respectively.

Table 4.1: **Unconditional Bias of the Estimators**

MODEL	$\hat{T}_{RT}$	$\hat{T}_{NW}$	$\hat{T}_{HT}$	$\hat{T}_R$
Linear	212.1953	935.7327	-16.70931	-16.08618
Quadratic	12.20103	568.9697	-30.06625	-31.18455
Exponential	-2.273007	-57.98402	-12.75688	-5.988498

4.4 The Mean Squared Error

The measures of the mean squared errors were computed for the four data sets and then compared.

$$MSE = \frac{\sum_{i=1}^{300}(\hat{T}_i - T)^2}{300} \tag{4.4}$$

The results are tabulated in Table 2 below,

Table 4.2: Mean Squared Error of the Estimators

MODEL	T_{RT}	$\hat{T}_{NW}$	$\hat{T}_{HT}$	$\hat{T}_{R}$
Linear	150.0895	2918.652	0.9306704	0.8625511
Quadratic	0.4962173	1079.088	3.013264	3.241588
Exponential	0.01722187	11.20716	0.5424597	0.1195404

4.5 Conditional Properties of the Estimator

Here, the samples selected are grouped into groups of size 20 therefore we have 15 groups. The grand mean for each group is computed as

$$\bar{\bar{X}} = \frac{1}{15}\sum_{i=1}^{20} \bar{x}_i \tag{4.5}$$

The mean estimator is also computed as

$$\hat{\bar{T}}_{TR} = \frac{1}{15}\sum_{i=1}^{20} \hat{T}_{TR\,i} \tag{4.6}$$

The conditional bias for each group was then computed as $(\hat{\bar{T}}_{TR} - \bar{Y})$ where $\bar{Y}$ is the population mean for the survey measurement and $\bar{x}_i$ is the sample mean for the auxiliary variables. The behavior of the conditional bias for each estimate is checked against each mean function.

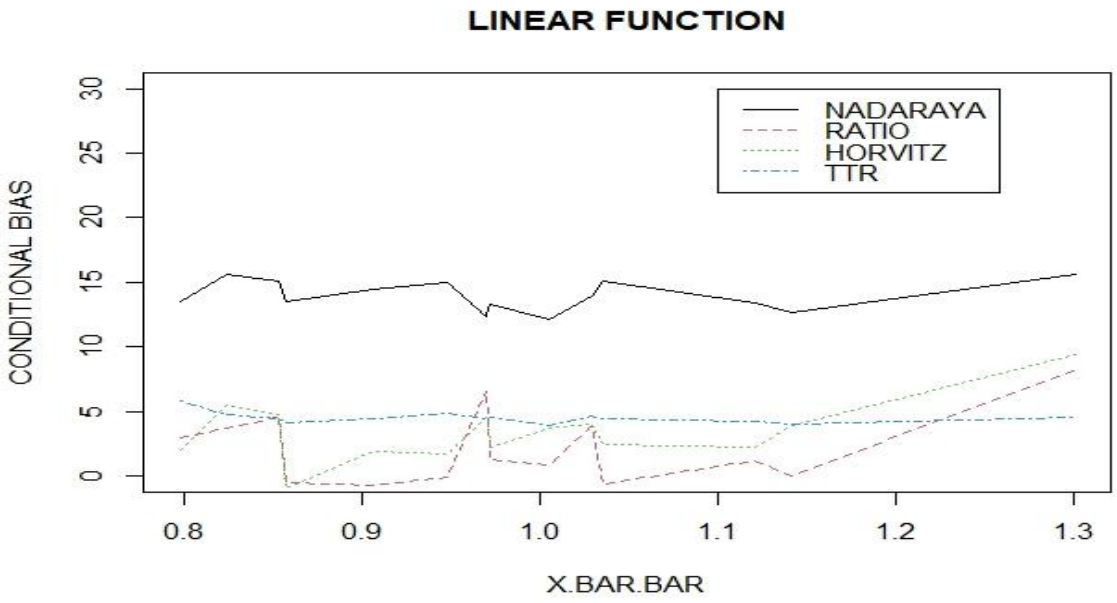

Figure 4.2: Conditional Bias for a Linear Function.

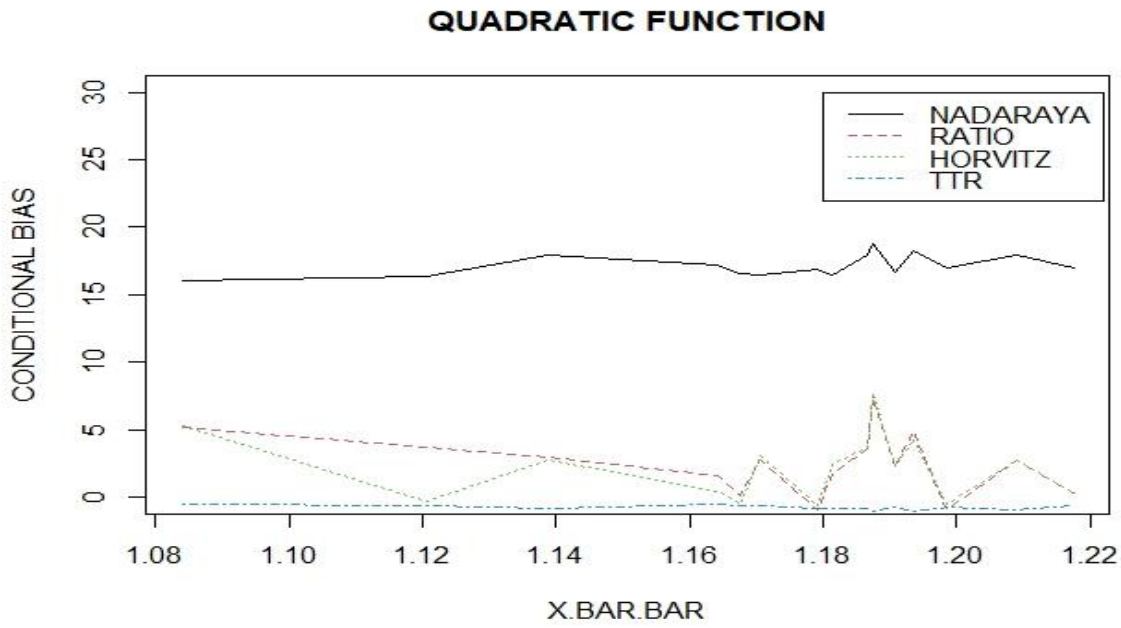

Figure 4.3: Conditional Bias for Quadratic Mean Function

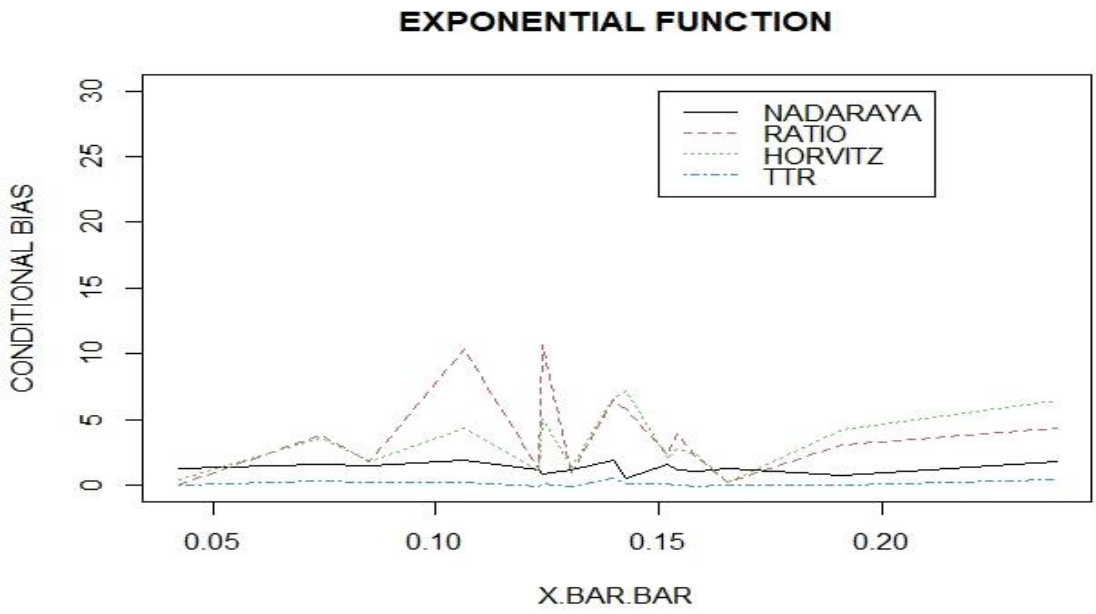

Figure 4.4: Conditional Bias for an Exponential Mean Function

CHAPTER FIVE

DISCUSSIONS

5.1 Introduction

This chapter presents the discussion of the results obtained after data was analyzed. Data was simulated and analyzed by R statistical package. The tabulated results and figures presented in chapter four are discussed in details.

5.2 Unconditional Bias

From Table 4.1, some of the values of the biases are negative and others are positive which indicate either underestimation or overestimation. For the linear function, the ratio estimator has the lowest bias followed -16.08618, by the Horvitz-Thompson estimator with -16.70931 and the proposed estimator is the third with 212.1953. In quadratic model, the proposed estimator performs the best with a bias of 12.20103. In exponential model, the proposed estimator has the lowest bias of -2.273007 which indicates that it's the best.

5.3 Unconditional MSE

From Table 4.2, the ratio estimator has the least MSE of 0.8625511 followed by the Horvitz-Thompson estimator with 0.9306704 under the linear function. For the quadratic function, the proposed estimator performed the best with a MSE of 0.4962173 followed by the Horvitz-Thompson estimator. For the exponential, the proposed estimator outperformed the other three models with a MSE of 0.01722187.

5.4 Conditional Properties

The figures 4.2, 4.3 and 4.4 in chapter four shows the trend of the conditional bias for each estimator under the three mean functions.

Figure 4.2 shows the conditional bias under the linear mean function. Figure 4.3 shows the conditional bias of the estimators under the quadratic mean function and figure 4.4 shows the conditional bias under the exponential mean function.

From figure 4.2, where the linear mean function was applied, the ratio estimator gave the best results. This is attributed to the fact that the ratio estimator is the Best Linear Unbiased Estimator (BLUE) thus it cannot be outperformed by any other estimator. It can be observed from the graph that the biases of the estimators are minimal.

Figure 4.3 where a quadratic mean function was applied, the proposed estimator gave the best estimates followed by the Horvitz-Thompson estimator, ratio estimator and the Nadaraya-Watson estimator performed poorly. It can also be observed from the graph that the bias between the estimators is large on the left but reduces towards the right as the mean increases.

From figure 4.4 the exponential mean function was applied, the proposed estimator gave better estimates of the population total followed by the Nadaraya-Watson estimator, the Horvitz-Thompson estimator and the ratio estimator performed poorly. It can be observed from the graph that the biases are minimal throughout the graph.

CHAPTER SIX

CONCLUSION, RECOMMENDATIONS AND PUBLICATION

6.1 Introduction

In this chapter, the summary of the work is given out in details. Also, the conclusions of the work are outlined as per the objectives of the study and the recommendations given.

6.2 Conclusion

A nonparametric estimator for a finite population total that addresses the problem of boundary bias effectively by the use of a hybrid of data transformation and data reflection techniques was proposed. The properties of the estimator were studied and applied on a simulated data. These include the unconditional biases and MSE's and the conditional biases and MSE's.

The first objective of the study was to propose a nonparametric estimator of finite population total. This study developed an estimator of finite population total based on a hybrid of data transformation and data reflection techniques which addressed the problem of boundary bias effectively as shown from the biases tabulated in Table 4.1. The proposed estimator was found to perform quite well under the quadratic and exponential models where it produced low biases as compared to the Ratio estimator, Horvitz-Thompson and the Nadaraya-Watson estimator. However, the ratio estimator was the best under linear models since it's the Best Linear Unbiased Estimator (BLUE). Our estimator has the least mean squared error over the two models. The conditional biases shown in figures 4.2 – 4.4 shows that the proposed estimator outperformed other estimators.

For investigation of properties of the proposed estimator to be carried out, simulation of data was done using R statistical software under the theoretical models. Table 4.2 shows the mean squared errors and the proposed estimator performed the best. It's therefore evident that the

proposed nonparametric estimator worked well in eliminating the boundary bias as compared to the Nadaraya-Watson, the Horvitz-Thompson estimators and the ratio estimator.

6.3 Recommendations

From this study, the proposed nonparametric estimator for a finite population total was developed and it performed better than (Dorfman, 1992) estimator and therefore can be recommended for estimation of finite population total and addressing the boundary problem. Estimation of the proposed estimator in this study was based on stratified random sampling. Estimation using cluster sampling ought to be carried out and the performance compared. The estimator was applied to a simulated data from linear, quadratic and exponential models therefore, further research should be carried out to improve the estimator in order to work on all the theoretical data variables.

6.4 Publication

Mugambi N., Odhiambo R. and Okungu J. (2023). Non-parametric estimator for a finite population total under stratified sampling incorporating a hybrid of data reflection and data transformation techniques. Journal of Mathematical Theory and Modelling ISSN 2224-5804 (Paper) ISSN 2225-0522 (Online) Volume 13, No.1 (2023) page 39-51.

REFERENCES

Bii, N. K., Onyango, C. O., & Odhiambo, J. (2020). *Estimating a finite population mean using transformed data in presence of random nonresponse*. International Journal of Mathematics and Mathematical Sciences, 2020

Bii, N. K., Onyango, C. O., & Odhiambo, J. (2019*). Boundary bias correction using weighting method in presence of nonresponse in two-stage cluster sampling*. Journal of Probability and Statistics, 2019.

Breidt, F. J., & Opsomer, J. D. (2000). *Local polynomial regression estimators in survey sampling*. Annals of statistics, 1026-1053.

Breidt, F. J., Claeskens, G., & Opsomer, J. D. (2005). *Model-assisted estimation for complex surveys using penalised splines*. Biometrika, 92(4), 831-846.

Chaudhuri, A., & Stenger, H. (2005). *Survey sampling*: theory and methods. CRC Press.

Cheruiyot, L. R. (2020). *Exploring Data-Reflection Technique in Nonparametric Regression Estimation of Finite Population* Total: An Empirical Study. American Journal of Theoretical and Applied Statistics, 9(4), 101-105.

Cheruiyot, L. R., Otieno, O. R., & Orwa, G. O. (2019). *A Boundary Corrected Non-Parametric Regression Estimator for Finite Population Total*. AL JOUR, 8(3), 83.

Cochran, W.G. (1977), *Sampling techniques, Third edition,* New York: John Wiley and Sons

Cormen, T. H., Leiserson, C. E., Rivest, R. L., & Stein, C. (2022). *Introduction to algorithms*. MIT press.

Cowling, A., & Hall, P. (1996). *On pseudodata methods for removing boundary effects in kernel density estimation.* Journal of the Royal Statistical Society: Series B (Methodological), 58(3), 551-563.

Dorfman, A. H. (1992). *Nonparametric regression for estimating totals in finite populations. In Proceedings of the Section on Survey Research Methods* (pp. 622-625). American Statistical Association Alexandria, VA.

Fay, R. E. and Herriot, R. A. (1979*). Estimates of income for small places: An application of JamesStein procedures to census data*. Journal of the American Statistical Association 74: 269–277.

Ghosh, M. and Rao, J. (1994). *Small arear estimation*: An appraisal. Statistical Science, 9:54–76

Härdle, W. (1994). *Applied Nonparametric Regression Analysis.* Cambridge: Cambridge

Hidiroglou, M. (2007). *Small-area estimation: Theory and practice*. In Proceedings of the Survey Research Methods Section (pp. 3445-3456).

Horvitz, D. G., & Thompson, D. J. (1952). *A generalization of sampling without replacement from a finite universe.* Journal of the American Statistical Association, 47(260), 663-685. https://doi.org/10.1080/01621459.1952.10483446

Karunamuni, R. J., & Alberts, T. (2005). *On boundary correction in kernel density estimation*. Statistical Methodology, 2(3), 191-212.

Karunamuni, R. J., & Alberts, T. (2006). *A locally adaptive transformation method of boundary correction in kernel density estimation.* Journal of Statistical Planning and Inference, 136(9), 2936-2960.

Kim, J. Y., Breidt, F. J., & Opsomer, J. D. (2003). *Nonparametric regression estimation of finite population totals under two-stage sampling.* preprint.

Kriegler, B. and Berk, R. (2010). *Small area estimation of the homeless in Los Angeles: An application of cost-sensitive stochastic gradient boosting.* The Annals of Applied Statistics 4: 1234–1255.

Lang'at, R. C. (2017). *Robust Estimation of Finite Population Total Incorporating Data-Reflection Technique in Nonparametric Regression* (Doctoral dissertation, COPAS, JKUAT).

Lohr, S. L. (2021). *Sampling: design and analysis*. Chapman and Hall/CRC.

Nadaraya, E. A. (1964). *On estimating regression*. Theory of Probability & Its Applications, 9(1), 141-142.

Ombui, T. (2012). *Robust estimation of finite population total using local polynomial regression.* Abstracts of postgraduate thesis.

Orwa, G. O., Otieno, R. O., & Mwita, P. N. (2010). *Nonparametric mixed ratio estimator for a finite population total in stratified sampling.*

Pfeffermann, D. (2002). *Small area estimation-new developments and directions*. International Statistical Review, 70(1), 125-143.

Pfeffermann, D. (2013). *New important developments in small area estimation*. Statistical Science, 28(1), 40-68.

Rao, J. N. K. (2003). *Small Area Estimation*. New York: Wiley

Rueda, M., & Sánchez-Borrego, I. R. (2009). *A predictive estimator of finite population mean using nonparametric regression.* Computational Statistics, 24(1), 1-14.

Schuster, E. F. (1985). *Incorporating support constraints into nonparametric estimators of densities.* Communications in Statistics-Theory and methods, 14(5), 1123-1136.

Syengo, C. K. (2018). *Local Polynomial Regression Estimator of the Finite Population Total under Stratified Random Sampling: A Model-Based Approach* (Doctoral dissertation, JKUAT-PAUSTI)

Tsybakov, A. B. (2008). *Introduction to nonparametric estimation*. Springer Science & Business Medi University Press.

Wand, M.P., Marron, J.S. and Ruppert, D. (1991). *Transformations in Density Estimation (with discussion).* Journal of the American Statistical Association, 86, 343-361.

Watson, G. S. (1964). *Smooth regression analysis.* Sankhyā: The Indian Journal of Statistics, Series A, 359-372.

Zheng, H., & Little, R. J. (2003). *Penalized spline model-based estimation of the finite populations total from probability-proportional-to-size samples*. Journal of official Statistics, .19(2), 99

APPENDIX 1: R CODES

```
 set.seed(20)
e<-rnorm(1000,0,1)
e
x<-runif(1000,0,1)
x
sum(x)
mx1<-1+2*(x-0.5)  #linear
mx1
mx2<-1+2*(x-0.5)^2  #quadratic
mx2
mx3<-exp(-8*x)  #exponential
mx3
y1<-mx1+e
y2<-mx2+e
y3<-mx3+e
sum(y1)
sum(y2)
sum(y3)
sum(mx1)
par(mfrow=c(2,2))
plot(x,mx1)
plot(x,mx2)
plot(x,mx3)
sindex=sample(1000,300) #selection of the sample
x.sample=x[sindex]
x.sample
xnosample<-setdiff(x,x.sample)
xnosample
   # transformation
g<-x^2+2*x+2
xnosample^2+2*xnosample+2
  #reflection
xnosamplereflect<-(-(xnosample^2+2*xnosample+2))
xnosamplereflect
sum(x)
sum(x.sample)
sindex=sample(1000,300)
mx1.sample=mx1[sindex]
mx1.sample
mx1nosample<-setdiff(mx1,mx1.sample)
mx1nosample
mx1nosample^2+2*mx1nosample+2
mx1nosamplereflect<-(-(mx1nosample^2+2*mx1nosample+2))
mx1nosamplereflect
sum(mx1nosamplereflect)
sum(mx1.sample)
y1=mx1+e
y1
sum(y1)
y1.sample<-y1[sindex]
y1.sample
sum(y1.sample)
y1nosample<-setdiff(y1,y1.sample)
```

```
y1nosample
sum(y1.sample)
n<-300
N<-1000
h<-0.535 #bandwidth
kx<-3/4*(1-x^2) #epanechnikov
TNW1<-sum(y1.sample)+sum((3/4*((1-(x.sample-mx1.sample)/h)^2))*y1.sample)
TNW1
TR1<-(sum(y1.sample)/sum(mx1.sample))*sum(mx1)
TR1
Pi<-n/N
THT1<-sum(y1.sample/n*N)
THT1
Ttr1<-sum(y1.sample)+1/(n*h)*sum((3/4*(1-(xnosample-mx1nosamplereflect)/h)^2)+(3/4*(1-
(xnosample+mx1nosamplereflect)/h)^2)*y1nosample)
Ttr1
sindex=sample(1000,300)
mx2.sample=mx2[sindex]
mx2.sample
mx2nosample<-setdiff(mx2,mx2.sample)
mx2nosample
mx2nosample^2+2*mx2nosample+2
mx2nosamplereflect<-(-(mx2nosample^2+2*mx2nosample+2))
mx2nosamplereflect
sum(mx2nosamplereflect)
y2<-mx2+e
sum(y2)
y2.sample<-y2[sindex]
y2.sample
y2nosample<-setdiff(y2,y2.sample)
y2nosample
sum(y2.sample)
TNW2<-sum(y2.sample)+sum(3/4*((1-(x.sample-mx2.sample)/h)^2)*y2.sample)
TNW2
TR2<-(sum(y2.sample)/sum(mx2.sample))*sum(mx2)
TR2
Pi<-n/N
THT2<-sum(y2.sample/n*N)
THT2
Ttr2<-sum(y2.sample)+1/(n*h)*sum((3/4*(1-(xnosample-mx2nosamplereflect)/h)^2)+(3/4*(1-
(xnosample+mx2nosamplereflect)/h)^2)*y2nosample)
Ttr2
sindex=sample(1000,300)
mx3.sample=mx3[sindex]
mx3.sample
mx3nosample<-setdiff(mx3,mx3.sample)
mx3nosample
mx3nosample^2+2*mx3nosample+2
mx3nosamplereflect<-(-(mx3nosample^2+2*mx3nosample+2))
mx3nosamplereflect
sum(mx3nosamplereflect)
y3
sum(y3)
y3.sample<-y3[sindex]
y3.sample
```

```
y3nosample<-setdiff(y3,y3.sample)
y3nosample
sum(y3.sample)
TNW3<-sum(y3.sample)+sum(3/4*((1-(x.sample-mx3.sample)/h)^2)*y3.sample)
TNW3
TR3<-(sum(y3.sample)/sum(mx3.sample))*sum(mx3)
TR3
Pi<-n/N
THT3<-sum(y3.sample/n*N)
THT3
Ttr3<-sum(y3.sample)+1/(n*h)*sum((3/4*(1-(xnosample-mx3nosamplereflect)/h)^2)+(3/4*(1-
(xnosample+mx3nosamplereflect)/h)^2)*y3nosample)
Ttr3
#bias
B1<-sum(TNW1-sum(y1))
B1
B1<-sum(TR1-sum(y1))
B1
B1<-sum(THT1-sum(y1))
B1
B1<-sum(Ttr1-sum(y1))
B1

B2<-sum(TNW2-sum(y2))
B2
B2<-sum(TR2-sum(y2))
B2
B2<-sum(THT2-sum(y2))
B2
B2<-sum(Ttr2-sum(y2))
B2

B3<-sum(TNW3-sum(y3))
B3
B3<-sum(TR3-sum(y3))
B3
B3<-sum(THT3-sum(y3))
B3
B3<-sum(Ttr3-sum(y3))
B3
#mse
m1<-sum((TNW1-sum(y1))^2)/300
m1
m1<-sum((TR1-sum(y1))^2)/300
m1
m1<-sum((THT1-sum(y1))^2)/300
m1
m1<-sum((Ttr1-sum(y1))^2)/300
m1
m2<-sum((TNW2-sum(y2))^2)/300
m2
m2<-sum((TR2-sum(y2))^2)/300
m2
m2<-sum((THT2-sum(y2))^2)/300
m2
```

```
m2<-sum((Ttr2-sum(y2))^2)/300
m2
m3<-sum((TNW3-sum(y3))^2)/300
m3
m3<-sum((TR3-sum(y3))^2)/300
m3
m3<-sum((THT3-sum(y3))^2)/300
m3
m3<-sum((Ttr3-sum(y3))^2)/300
m3
sindex<-sample(1000,300)
ep<-e[sindex]
ep
M<-x.sample
M
m<-length(M)
m
sindex<-sample(300,20)
M.sample<-M[sindex]
M.sample
Mnosample<-setdiff(M,M.sample)
Mnosample
mM1<-1+2*(M-0.5)  #Linear
mM1
sum(mM1)
l<-length(mM1)
l
mM2<-1+2*(M-0.5)^2  #Quadratic
mM2
q<-length(mM2)
q
mM3<-exp(-8*M)
mM3
j<-length(mM3)
j
yL<-mM1+ep
yL
sum(yL)
mean(yL)
yQ<-mM2+ep
yQ
sum(yQ)
mean(yQ)
yE<-mM3+ep
yE
sum(yE)
mean(yE)
#Conditional bias for Linear Function
idx<-sample(rep(1:15,each=ceiling(l/15)),replace=FALSE)
A1<-mM1[idx==1]
A1
sum(A1)
mean(A1)
A1nosample<-setdiff(mM1,A1)
A1nosample
```

```
A1nosample^2+2*A1nosample+2
A1nosamplereflect<-(-(A1nosample^2+2*A1nosample+2))
A1nosamplereflect
sum(A1nosamplereflect)
yL.sample<-yL[idx==1]
yL.sample
sum(yL.sample)
yLnosample<-setdiff(yL,yL.sample)
yLnosample
n1<-20
N1<-300
h1<-1.35
TNWL1<-sum(yL.sample)+sum((3/4*((1-(M.sample-A1)/h1)^2))*yL.sample)
TNWL1
TRL1<-(sum(yL.sample)/sum(A1))*sum(mM1)
TRL1
Pi<-n1/N1
THTL1<-sum(yL.sample/n1*N1)
THTL1
TtrL1<-sum(yL.sample)+1/(n1*h1)*sum((3/4*(1-(Mnosample-A1nosamplereflect)/h1)^2)+(3/4*(1-
(Mnosample+A1nosamplereflect)/h1)^2)*yLnosample)
TtrL1
A2<-mM1[idx==2]
A2
mean(A2)
A2nosample<-setdiff(mM1,A2)
A2nosample
A2nosample^2+2*A2nosample+2
A2nosamplereflect<-(-(A2nosample^2+2*A2nosample+2))
A2nosamplereflect
yL.sample<-yL[idx==2]
yL.sample
yLnosample<-setdiff(yL,yL.sample)
yLnosample
TNWL2<-sum(yL.sample)+sum((3/4*((1-(M.sample-A2)/h1)^2))*yL.sample)
TNWL2
TRL2<-(sum(yL.sample)/sum(A2))*sum(mM1)
TRL2
Pi<-n1/N1
THTL2<-sum(yL.sample/n1*N1)
THTL2
TtrL2<-sum(yL.sample)+1/(n1*h1)*sum((3/4*(1-(Mnosample-A2nosamplereflect)/h1)^2)+(3/4*(1-
(Mnosample+A2nosamplereflect)/h1)^2)*yLnosample)
TtrL2
A3<-mM1[idx==3]
A3
mean(A3)
A3nosample<-setdiff(mM1,A3)
A3nosample
A3nosample^2+2*A3nosample+2
A3nosamplereflect<-(-(A3nosample^2+2*A3nosample+2))
A3nosamplereflect
yL.sample<-yL[idx==3]
yL.sample
yLnosample<-setdiff(yL,yL.sample)
```

```
yLnosample
TNWL3<-sum(yL.sample)+sum((3/4*((1-(M.sample-A3)/h1)^2))*yL.sample)
TNWL3
TRL3<-(sum(yL.sample)/sum(A3))*sum(mM1)
TRL3
Pi<-n1/N1
THTL3<-sum(yL.sample/n1*N1)
THTL3
TtrL3<-sum(yL.sample)+1/(n1*h1)*sum((3/4*(1-(Mnosample-A3nosamplereflect)/h1)^2)+(3/4*(1-
(Mnosample+A3nosamplereflect)/h1)^2)*yLnosample)
TtrL3
A4<-mM1[idx==4]
A4
mean(A4)
A4nosample<-setdiff(mM1,A4)
A4nosample
A4nosample^2+2*A4nosample+2
A4nosamplereflect<-(-(A4nosample^2+2*A4nosample+2))
A4nosamplereflect
yL.sample<-yL[idx==4]
yL.sample
yLnosample<-setdiff(yL,yL.sample)
yLnosample
TNWL4<-sum(yL.sample)+sum((3/4*((1-(M.sample-A4)/h1)^2))*yL.sample)
TNWL4
TRL4<-(sum(yL.sample)/sum(A4))*sum(mM1)
TRL4
Pi<-n1/N1
THTL4<-sum(yL.sample/n1*N1)
THTL4
TtrL4<-sum(yL.sample)+1/(n1*h1)*sum((3/4*(1-(Mnosample-A4nosamplereflect)/h1)^2)+(3/4*(1-
(Mnosample+A4nosamplereflect)/h1)^2)*yLnosample)
TtrL4
A5<-mM1[idx==5]
A5
mean(A5)
A5nosample<-setdiff(mM1,A5)
A5nosample
A5nosample^2+2*A5nosample+2
A5nosamplereflect<-(-(A5nosample^2+2*A5nosample+2))
A5nosamplereflect
yL.sample<-yL[idx==5]
yL.sample
yLnosample<-setdiff(yL,yL.sample)
yLnosample
TNWL5<-sum(yL.sample)+sum((3/4*((1-(M.sample-A5)/h1)^2))*yL.sample)
TNWL5
TRL5<-(sum(yL.sample)/sum(A5))*sum(mM1)
TRL5
Pi<-n1/N1
THTL5<-sum(yL.sample/n1*N1)
THTL5
TtrL5<-sum(yL.sample)+1/(n1*h1)*sum((3/4*(1-(Mnosample-A5nosamplereflect)/h1)^2)+(3/4*(1-
(Mnosample+A5nosamplereflect)/h1)^2)*yLnosample)
TtrL5
```

```
A6<-mM1[idx==6]
A6
mean(A6)
A6nosample<-setdiff(mM1,A6)
A6nosample
A6nosample^2+2*A6nosample+2
A6nosamplereflect<-(-(A6nosample^2+2*A6nosample+2))
A6nosamplereflect
yL.sample<-yL[idx==6]
yL.sample
yLnosample<-setdiff(yL,yL.sample)
yLnosample
TNWL6<-sum(yL.sample)+sum((3/4*((1-(M.sample-A6)/h1)^2))*yL.sample)
TNWL6
TRL6<-(sum(yL.sample)/sum(A6))*sum(mM1)
TRL6
Pi<-n1/N1
THTL6<-sum(yL.sample/n1*N1)
THTL6
TtrL6<-sum(yL.sample)+1/(n1*h)*sum((3/4*(1-(Mnosample-A6nosamplereflect)/h1)^2)+(3/4*(1-
(Mnosample+A6nosamplereflect)/h1)^2)*yLnosample)
TtrL6
A7<-mM1[idx==7]
A7
mean(A7)
A7nosample<-setdiff(mM1,A7)
A7nosample
A7nosample^2+2*A7nosample+2
A7nosamplereflect<-(-(A7nosample^2+2*A7nosample+2))
A7nosamplereflect
yL.sample<-yL[idx==7]
yL.sample
yLnosample<-setdiff(yL,yL.sample)
yLnosample
TNWL7<-sum(yL.sample)+sum((3/4*((1-(M.sample-A7)/h1)^2))*yL.sample)
TNWL7
TRL7<-(sum(yL.sample)/sum(A7))*sum(mM1)
TRL7
Pi<-n1/N1
THTL7<-sum(yL.sample/n1*N1)
THTL7
TtrL7<-sum(yL.sample)+1/(n1*h1)*sum((3/4*(1-(Mnosample-A7nosamplereflect)/h1)^2)+(3/4*(1-
(Mnosample+A7nosamplereflect)/h1)^2)*yLnosample)
TtrL7
A8<-mM1[idx==8]
A8
mean(A8)
A8nosample<-setdiff(mM1,A8)
A8nosample
A8nosample^2+2*A1nosample+2
A8nosamplereflect<-(-(A8nosample^2+2*A8nosample+2))
A8nosamplereflect
yL.sample<-yL[idx==8]
yL.sample
yLnosample<-setdiff(yL,yL.sample)
```

```
yLnosample
TNWL8<-sum(yL.sample)+sum((3/4*((1-(M.sample-A8)/h1)^2))*yL.sample)
TNWL8
TRL8<-(sum(yL.sample)/sum(A8))*sum(mM1)
TRL8
Pi<-n1/N1
THTL8<-sum(yL.sample/n1*N1)
THTL8
TtrL8<-sum(yL.sample)+1/(n1*h1)*sum((3/4*(1-(Mnosample-A8nosamplereflect)/h1)^2)+(3/4*(1-
(Mnosample+A8nosamplereflect)/h1)^2)*yLnosample)
TtrL8
A9<-mM1[idx==9]
A9
mean(A9)
A9nosample<-setdiff(mM1,A9)
A9nosample
A9nosample^2+2*A9nosample+2
A9nosamplereflect<-(-(A9nosample^2+2*A9nosample+2))
A9nosamplereflect
yL.sample<-yL[idx==9]
yL.sample
yLnosample<-setdiff(yL,yL.sample)
yLnosample
TNWL9<-sum(yL.sample)+sum((3/4*((1-(M.sample-A9)/h1)^2))*yL.sample)
TNWL9
TRL9<-(sum(yL.sample)/sum(A9))*sum(mM1)
TRL9
Pi<-n1/N1
THTL9<-sum(yL.sample/n1*N1)
THTL9
TtrL9<-sum(yL.sample)+1/(n1*h1)*sum((3/4*(1-(Mnosample-A9nosamplereflect)/h1)^2)+(3/4*(1-
(Mnosample+A9nosamplereflect)/h1)^2)*yLnosample)
TtrL9

A10<-mM1[idx==10]
A10
mean(A10)
A10nosample<-setdiff(mM1,A10)
A10nosample
A10nosample^2+2*A10nosample+2
A10nosamplereflect<-(-(A10nosample^2+2*A10nosample+2))
A10nosamplereflect
yL.sample<-yL[idx==10]
yL.sample
yLnosample<-setdiff(yL,yL.sample)
yLnosample
TNWL10<-sum(yL.sample)+sum((3/4*((1-(M.sample-A10)/h1)^2))*yL.sample)
TNWL10
TRL10<-(sum(yL.sample)/sum(A10))*sum(mM1)
TRL10
Pi<-n1/N1
THTL10<-sum(yL.sample/n1*N1)
THTL10
TtrL10<-sum(yL.sample)+1/(n1*h1)*sum((3/4*(1-(Mnosample-A10nosamplereflect)/h1)^2)+(3/4*(1-
(Mnosample+A10nosamplereflect)/h1)^2)*yLnosample)
```

```
TtrL10
A11<-mM1[idx==11]
A11
mean(A11)
A11nosample<-setdiff(mM1,A11)
A11nosample
A11nosample^2+2*A11nosample+2
A11nosamplereflect<-(-(A11nosample^2+2*A11nosample+2))
A11nosamplereflect
yL.sample<-yL[idx==11]
yL.sample
yLnosample<-setdiff(yL,yL.sample)
yLnosample
TNWL11<-sum(yL.sample)+sum((3/4*((1-(M.sample-A11)/h1)^2))*yL.sample)
TNWL11
TRL11<-(sum(yL.sample)/sum(A11))*sum(mM1)
TRL11
Pi<-n1/N1
THTL11<-sum(yL.sample/n1*N1)
THTL11
TtrL11<-sum(yL.sample)+1/(n1*h1)*sum((3/4*(1-(Mnosample-A11nosamplereflect)/h1)^2)+(3/4*(1-
(Mnosample+A11nosamplereflect)/h1)^2)*yLnosample)
TtrL11
A12<-mM1[idx==12]
A12
mean(A12)
A12nosample<-setdiff(mM1,A12)
A12nosample
A12nosample^2+2*A12nosample+2
A12nosamplereflect<-(-(A12nosample^2+2*A12nosample+2))
A12nosamplereflect
yL.sample<-yL[idx==12]
yL.sample
yLnosample<-setdiff(yL,yL.sample)
yLnosample
TNWL12<-sum(yL.sample)+sum((3/4*((1-(M.sample-A12)/h1)^2))*yL.sample)
TNWL12
TRL12<-(sum(yL.sample)/sum(A12))*sum(mM1)
TRL12
Pi<-n1/N1
THTL12<-sum(yL.sample/n1*N1)
THTL12
TtrL12<-sum(yL.sample)+1/(n1*h1)*sum((3/4*(1-(Mnosample-A12nosamplereflect)/h1)^2)+(3/4*(1-
(Mnosample+A12nosamplereflect)/h1)^2)*yLnosample)
TtrL12
A13<-mM1[idx==13]
A13
mean(A13)
A13nosample<-setdiff(mM1,A13)
A13nosample
A13nosample^2+2*A13nosample+2
A13nosamplereflect<-(-(A13nosample^2+2*A13nosample+2))
A13nosamplereflect
yL.sample<-yL[idx==13]
yL.sample
```

```
yLnosample<-setdiff(yL,yL.sample)
yLnosample
TNWL13<-sum(yL.sample)+sum((3/4*((1-(M.sample-A13)/h1)^2))*yL.sample)
TNWL13
TRL13<-(sum(yL.sample)/sum(A13))*sum(mM1)
TRL13
Pi<-n1/N1
THTL13<-sum(yL.sample/n1*N1)
THTL13
TtrL13<-sum(yL.sample)+1/(n1*h1)*sum((3/4*(1-(Mnosample-A13nosamplereflect)/h1)^2)+(3/4*(1-
(Mnosample+A13nosamplereflect)/h1)^2)*yLnosample)
TtrL13
A14<-mM1[idx==14]
A14
mean(A14)
A14nosample<-setdiff(mM1,A14)
A14nosample
A14nosample^2+2*A14nosample+2
A14nosamplereflect<-(-(A14nosample^2+2*A14nosample+2))
A14nosamplereflect
yL.sample<-yL[idx==14]
yL.sample
yLnosample<-setdiff(yL,yL.sample)
yLnosample
TNWL14<-sum(yL.sample)+sum((3/4*((1-(M.sample-A14)/h1)^2))*yL.sample)
TNWL14
TRL1<-(sum(yL.sample)/sum(A14))*sum(mM1)
TRL1
Pi<-n1/N1
THTL14<-sum(yL.sample/n1*N1)
THTL14
TtrL14<-sum(yL.sample)+1/(n1*h1)*sum((3/4*(1-(Mnosample-A14nosamplereflect)/h1)^2)+(3/4*(1-
(Mnosample+A14nosamplereflect)/h1)^2)*yLnosample)
TtrL14
A15<-mM1[idx==15]
A15
mean(A15)
A15nosample<-setdiff(mM1,A15)
A15nosample
A15nosample^2+2*A15nosample+2
A15nosamplereflect<-(-(A15nosample^2+2*A15nosample+2))
A15nosamplereflect
yL.sample<-yL[idx==15]
yL.sample
yLnosample<-setdiff(yL,yL.sample)
yLnosample
TNWL15<-sum(yL.sample)+sum((3/4*((1-(M.sample-A15)/h1)^2))*yL.sample)
TNWL15
TRL15<-(sum(yL.sample)/sum(A15))*sum(mM1)
TRL15
Pi<-n1/N1
THTL15<-sum(yL.sample/n1*N1)
THTL15
TtrL15<-sum(yL.sample)+1/(n1*h1)*sum((3/4*(1-(Mnosample-A15nosamplereflect)/h1)^2)+(3/4*(1-
(Mnosample+A15nosamplereflect)/h1)^2)*yLnosample)
```

```
TtrL15
#conditional bias for quadratic function
idx<-sample(rep(1:15,each=floor(q/15)),replace=FALSE)
C1<-mM2[idx==1]
C1
mean(C1)
C1nosample<-setdiff(mM2,C1)
C1nosample
C1nosample^2+2*C1nosample+2
C1nosamplereflect<-(-(C1nosample^2+2*C1nosample+2))
C1nosamplereflect
yQ.sample<-yQ[idx==1]
yQ.sample
yQnosample<-setdiff(yQ,yQ.sample)
yQnosample
TNWQ1<-sum(yQ.sample)+sum((3/4*((1-(M.sample-C1)/h1)^2))*yQ.sample)
TNWQ1
TRQ1<-(sum(yQ.sample)/sum(C1))*sum(mM2)
TRQ1
Pi<-n1/N1
THTQ1<-sum(yQ.sample/n1*N1)
THTQ1
TtrQ1<-sum(yQ.sample)+1/(n1*h1)*sum((3/4*(1-(Mnosample-C1nosamplereflect)/h1)^2)+(3/4*(1-
(Mnosample+C1nosamplereflect)/h1)^2)*yQnosample)
TtrQ1
C2<-mM2[idx==2]
C2
mean(C2)
C2nosample<-setdiff(mM2,C2)
C2nosample
C2nosample^2+2*C2nosample+2
C2nosamplereflect<-(-(C2nosample^2+2*C2nosample+2))
C2nosamplereflect
yQ.sample<-yQ[idx==2]
yQ.sample
yQnosample<-setdiff(yQ,yQ.sample)
yQnosample
TNWQ2<-sum(yQ.sample)+sum((3/4*((1-(M.sample-C2)/h1)^2))*yQ.sample)
TNWQ2
TRQ2<-(sum(yQ.sample)/sum(C2))*sum(mM2)
TRQ2
Pi<-n1/N1
THTQ2<-sum(yL.sample/n1*N1)
THTQ2
TtrQ2<-sum(yQ.sample)+1/(n1*h1)*sum((3/4*(1-(Mnosample-C2nosamplereflect)/h1)^2)+(3/4*(1-
(Mnosample+C2nosamplereflect)/h1)^2)*yQnosample)
TtrQ2
C3<-mM2[idx==3]
C3
mean(C3)
C3nosample<-setdiff(mM2,C3)
C3nosample
C3nosample^2+2*C3nosample+2
C3nosamplereflect<-(-(C3nosample^2+2*C3nosample+2))
C3nosamplereflect
```

```
yQ.sample<-yQ[idx==3]
yQ.sample
yQnosample<-setdiff(yQ,yQ.sample)
yQnosample
TNWQ3<-sum(yQ.sample)+sum((3/4*((1-(M.sample-C3)/h1)^2))*yQ.sample)
TNWQ3
TRQ3<-(sum(yQ.sample)/sum(C3))*sum(mM2)
TRQ3
Pi<-n1/N1
THTQ3<-sum(yQ.sample/n1*N1)
THTQ3
TtrQ3<-sum(yQ.sample)+1/(n1*h1)*sum((3/4*(1-(Mnosample-C3nosamplereflect)/h1)^2)+(3/4*(1-
(Mnosample+C3nosamplereflect)/h1)^2)*yQnosample)
TtrQ3
C4<-mM2[idx==4]
C4
mean(C4)
C4nosample<-setdiff(mM2,C4)
C4nosample
C4nosample^2+2*C4nosample+2
C4nosamplereflect<-(-(C4nosample^2+2*C4nosample+2))
C4nosamplereflect
yQ.sample<-yQ[idx==4]
yQ.sample
yQnosample<-setdiff(yQ,yQ.sample)
yQnosample
TNWQ4<-sum(yQ.sample)+sum((3/4*((1-(M.sample-C4)/h1)^2))*yQ.sample)
TNWQ4
TRQ4<-(sum(yQ.sample)/sum(C4))*sum(mM2)
TRQ4
Pi<-n1/N1
THTQ4<-sum(yQ.sample/n1*N1)
THTQ4
TtrQ4<-sum(yQ.sample)+1/(n1*h1)*sum((3/4*(1-(Mnosample-C4nosamplereflect)/h1)^2)+(3/4*(1-
(Mnosample+C4nosamplereflect)/h1)^2)*yQnosample)
TtrQ4
C5<-mM2[idx==5]
C5
mean(C5)
C5nosample<-setdiff(mM2,C5)
C5nosample
C5nosample^2+2*C5nosample+2
C5nosamplereflect<-(-(C5nosample^2+2*C5nosample+2))
C5nosamplereflect
yQ.sample<-yQ[idx==5]
yQ.sample
yQnosample<-setdiff(yQ,yQ.sample)
yQnosample
TNWQ5<-sum(yQ.sample)+sum((3/4*((1-(M.sample-C5)/h1)^2))*yQ.sample)
TNWQ5
TRQ5<-(sum(yQ.sample)/sum(C5))*sum(mM2)
TRQ5
Pi<-n1/N1
THTQ5<-sum(yQ.sample/n1*N1)
THTQ5
```

```
TtrQ5<-sum(yQ.sample)+1/(n1*h1)*sum((3/4*(1-(Mnosample-C5nosamplereflect)/h1)^2)+(3/4*(1-
(Mnosample+C5nosamplereflect)/h1)^2)*yQnosample)
TtrQ5
C6<-mM2[idx==6]
C6
mean(C6)
C6nosample<-setdiff(mM2,C6)
C6nosample
C6nosample^2+2*C6nosample+2
C6nosamplereflect<-(-(C6nosample^2+2*C6nosample+2))
C6nosamplereflect
yQ.sample<-yQ[idx==6]
yQ.sample
yQnosample<-setdiff(yQ,yQ.sample)
yQnosample
TNWQ6<-sum(yQ.sample)+sum((3/4*((1-(M.sample-C6)/h1)^2))*yQ.sample)
TNWQ6
TRQ6<-(sum(yQ.sample)/sum(C6))*sum(mM2)
TRQ6
Pi<-n1/N1
THTQ6<-sum(yQ.sample/n1*N1)
THTQ6
TtrQ6<-sum(yQ.sample)+1/(n1*h1)*sum((3/4*(1-(Mnosample-C6nosamplereflect)/h1)^2)+(3/4*(1-
(Mnosample+C6nosamplereflect)/h1)^2)*yQnosample)
TtrQ6
C7<-mM2[idx==7]
C7
mean(C7)
C7nosample<-setdiff(mM2,C7)
C7nosample
C7nosample^2+2*C7nosample+2
C7nosamplereflect<-(-(C7nosample^2+2*C7nosample+2))
C7nosamplereflect
yQ.sample<-yQ[idx==7]
yQ.sample
yQnosample<-setdiff(yQ,yQ.sample)
yQnosample
TNWQ7<-sum(yQ.sample)+sum((3/4*((1-(M.sample-C7)/h1)^2))*yQ.sample)
TNWQ7
TRQ7<-(sum(yQ.sample)/sum(C7))*sum(mM2)
TRQ7
Pi<-n1/N1
THTQ7<-sum(yQ.sample/n1*N1)
THTQ7
TtrQ7<-sum(yQ.sample)+1/(n1*h1)*sum((3/4*(1-(Mnosample-C7nosamplereflect)/h1)^2)+(3/4*(1-
(Mnosample+C7nosamplereflect)/h1)^2)*yQnosample)
TtrQ7
C8<-mM2[idx==8]
C8
mean(C8)
C8nosample<-setdiff(mM2,C8)
C8nosample
C8nosample^2+2*C8nosample+2
C8nosamplereflect<-(-(C8nosample^2+2*C8nosample+2))
C8nosamplereflect
```

```
yQ.sample<-yQ[idx==8]
yQ.sample
yQnosample<-setdiff(yQ,yQ.sample)
yQnosample
TNWQ8<-sum(yQ.sample)+sum((3/4*((1-(M.sample-C8)/h1)^2))*yQ.sample)
TNWQ8
TRQ8<-(sum(yQ.sample)/sum(C8))*sum(mM2)
TRQ8
Pi<-n1/N1
THTQ8<-sum(yQ.sample/n1*N1)
THTQ8
TtrQ8<-sum(yQ.sample)+1/(n1*h1)*sum((3/4*(1-(Mnosample-C8nosamplereflect)/h1)^2)+(3/4*(1-
(Mnosample+C8nosamplereflect)/h1)^2)*yQnosample)
TtrQ8
C9<-mM2[idx==9]
C9
mean(C9)
C9nosample<-setdiff(mM2,C9)
C9nosample
C9nosample^2+2*C9nosample+2
C9nosamplereflect<-(-(C9nosample^2+2*C9nosample+2))
C9nosamplereflect
yQ.sample<-yQ[idx==9]
yQ.sample
yQnosample<-setdiff(yQ,yQ.sample)
yQnosample
TNWQ9<-sum(yQ.sample)+sum((3/4*((1-(M.sample-C9)/h1)^2))*yQ.sample)
TNWQ9
TRQ9<-(sum(yQ.sample)/sum(C9))*sum(mM2)
TRQ9
Pi<-n1/N1
THTQ9<-sum(yQ.sample/n1*N1)
THTQ9
TtrQ9<-sum(yQ.sample)+1/(n1*h1)*sum((3/4*(1-(Mnosample-C9nosamplereflect)/h1)^2)+(3/4*(1-
(Mnosample+C9nosamplereflect)/h1)^2)*yQnosample)
TtrQ9
C10<-mM2[idx==10]
C10
mean(C10)
C10nosample<-setdiff(mM2,C10)
C10nosample
C10nosample^2+2*C10nosample+2
C10nosamplereflect<-(-(C10nosample^2+2*C10nosample+2))
C10nosamplereflect
yQ.sample<-yQ[idx==10]
yQ.sample
yQnosample<-setdiff(yQ,yQ.sample)
yQnosample
TNWQ10<-sum(yQ.sample)+sum((3/4*((1-(M.sample-C10)/h1)^2))*yQ.sample)
TNWQ10
TRQ10<-(sum(yQ.sample)/sum(C10))*sum(mM2)
TRQ10
Pi<-n1/N1
THTQ10<-sum(yQ.sample/n1*N1)
THTQ10
```

```
TtrQ10<-sum(yQ.sample)+1/(n1*h1)*sum((3/4*(1-(Mnosample-C10nosamplereflect)/h1)^2)+(3/4*(1-
(Mnosample+C10nosamplereflect)/h1)^2)*yQnosample)
TtrQ10
C11<-mM2[idx==11]
C11
mean(C11)
C11nosample<-setdiff(mM2,C11)
C11nosample
C11nosample^2+2*C11nosample+2
C11nosamplereflect<-(-(C11nosample^2+2*C11nosample+2))
C11nosamplereflect
yQ.sample<-yQ[idx==11]
yQ.sample
yQnosample<-setdiff(yQ,yQ.sample)
yQnosample
TNWQ11<-sum(yQ.sample)+sum((3/4*((1-(M.sample-C11)/h1)^2))*yQ.sample)
TNWQ11
TRQ11<-(sum(yQ.sample)/sum(C11))*sum(mM2)
TRQ11
Pi<-n1/N1
THTQ11<-sum(yQ.sample/n1*N1)
THTQ11
TtrQ11<-sum(yQ.sample)+1/(n1*h1)*sum((3/4*(1-(Mnosample-C11nosamplereflect)/h1)^2)+(3/4*(1-
(Mnosample+C11nosamplereflect)/h1)^2)*yQnosample)
TtrQ11
C12<-mM2[idx==12]
C12
mean(C12)
C12nosample<-setdiff(mM2,C12)
C12nosample
C12nosample^2+2*C12nosample+2
C12nosamplereflect<-(-(C12nosample^2+2*C12nosample+2))
C12nosamplereflect
yQ.sample<-yQ[idx==12]
yQ.sample
yQnosample<-setdiff(yQ,yQ.sample)
yQnosample
TNWQ12<-sum(yQ.sample)+sum((3/4*((1-(M.sample-C12)/h1)^2))*yQ.sample)
TNWQ12
TRQ12<-(sum(yQ.sample)/sum(C12))*sum(mM2)
TRQ12
Pi<-n1/N1
THTQ12<-sum(yQ.sample/n1*N1)
THTQ12
TtrQ12<-sum(yQ.sample)+1/(n1*h1)*sum((3/4*(1-(Mnosample-C12nosamplereflect)/h1)^2)+(3/4*(1-
(Mnosample+C12nosamplereflect)/h1)^2)*yQnosample)
TtrQ12
C13<-mM2[idx==13]
C13
mean(C13)
C13nosample<-setdiff(mM2,C13)
C13nosample
C13nosample^2+2*C13nosample+2
C13nosamplereflect<-(-(C13nosample^2+2*C13nosample+2))
C13nosamplereflect
```

```
yQ.sample<-yQ[idx==13]
yQ.sample
yQnosample<-setdiff(yQ,yQ.sample)
yQnosample
TNWQ13<-sum(yQ.sample)+sum((3/4*((1-(M.sample-C13)/h1)^2))*yQ.sample)
TNWQ13
TRQ13<-(sum(yQ.sample)/sum(C13))*sum(mM2)
TRQ13
Pi<-n1/N1
THTQ13<-sum(yQ.sample/n1*N1)
THTQ13
TtrQ13<-sum(yQ.sample)+1/(n1*h1)*sum((3/4*(1-(Mnosample-C13nosamplereflect)/h1)^2)+(3/4*(1-
(Mnosample+C13nosamplereflect)/h1)^2)*yQnosample)
TtrQ13
C14<-mM2[idx==14]
C14
mean(C14)
C14nosample<-setdiff(mM2,C14)
C14nosample
C14nosample^2+2*C14nosample+2
C14nosamplereflect<-(-(C14nosample^2+2*C14nosample+2))
C14nosamplereflect
yQ.sample<-yQ[idx==14]
yQ.sample
yQnosample<-setdiff(yQ,yQ.sample)
yQnosample
TNWQ14<-sum(yQ.sample)+sum((3/4*((1-(M.sample-C14)/h1)^2))*yQ.sample)
TNWQ14
TRQ14<-(sum(yQ.sample)/sum(C14))*sum(mM2)
TRQ14
Pi<-n1/N1
THTQ14<-sum(yQ.sample/n1*N1)
THTQ14
TtrQ14<-sum(yQ.sample)+1/(n1*h1)*sum((3/4*(1-(Mnosample-C14nosamplereflect)/h1)^2)+(3/4*(1-
(Mnosample+C14nosamplereflect)/h1)^2)*yQnosample)
TtrQ14
C15<-mM2[idx==15]
C15
mean(C15)
C15nosample<-setdiff(mM2,C15)
C15nosample
C15nosample^2+2*C15nosample+2
C15nosamplereflect<-(-(C15nosample^2+2*C15nosample+2))
C15nosamplereflect
yQ.sample<-yQ[idx==15]
yQ.sample
yQnosample<-setdiff(yQ,yQ.sample)
yQnosample
TNWQ15<-sum(yQ.sample)+sum((3/4*((1-(M.sample-C15)/h1)^2))*yQ.sample)
TNWQ15
TRQ15<-(sum(yQ.sample)/sum(C15))*sum(mM2)
TRQ15
Pi<-n1/N1
THTQ15<-sum(yQ.sample/n1*N1)
THTQ15
```

```
TtrQ15<-sum(yQ.sample)+1/(n1*h1)*sum((3/4*(1-(Mnosample-C15nosamplereflect)/h1)^2)+(3/4*(1-
(Mnosample+C15nosamplereflect)/h1)^2)*yQnosample)
TtrQ15
 #Coditional bias for exponential function
idx<-sample(rep(1:15,each=floor(j/15)),replace=FALSE)
D1<-mM3[idx==1]
D1
mean(D1)
D1nosample<-setdiff(mM3,D1)
D1nosample
D1nosample^2+2*D1nosample+2
D1nosamplereflect<-(-(D1nosample^2+2*D1nosample+2))
D1nosamplereflect
yE.sample<-yE[idx==1]
yE.sample
yEnosample<-setdiff(yE,yE.sample)
yEnosample
TNWE1<-sum(yE.sample)+sum((3/4*((1-(M.sample-D1)/h1)^2))*yE.sample)
TNWE1
TRE1<-(sum(yE.sample)/sum(D1))*sum(mM3)
TRE1
Pi<-n1/N1
THTE1<-sum(yE.sample/n1*N1)
THTE1
TtrE1<-sum(yE.sample)+1/(n1*h1)*sum((3/4*(1-(Mnosample-D1nosamplereflect)/h1)^2)+(3/4*(1-
(Mnosample+D1nosamplereflect)/h1)^2)*yEnosample)
TtrE1
D2<-mM3[idx==2]
D2
mean(D2)
D2nosample<-setdiff(mM3,D2)
D2nosample
D2nosample^2+2*D2nosample+2
D2nosamplereflect<-(-(D2nosample^2+2*D2nosample+2))
D2nosamplereflect
yE.sample<-yE[idx==2]
yE.sample
yEnosample<-setdiff(yE,yE.sample)
yEnosample
TNWE2<-sum(yE.sample)+sum((3/4*((1-(M.sample-D2)/h1)^2))*yE.sample)
TNWE2
TRE2<-(sum(yE.sample)/sum(D2))*sum(mM3)
TRE2
Pi<-n1/N1
THTE2<-sum(yE.sample/n1*N1)
THTE2
TtrE2<-sum(yE.sample)+1/(n1*h1)*sum((3/4*(1-(Mnosample-D2nosamplereflect)/h1)^2)+(3/4*(1-
(Mnosample+D2nosamplereflect)/h1)^2)*yEnosample)
TtrE2
D3<-mM3[idx==3]
D3
mean(D3)
D3nosample<-setdiff(mM3,D3)
D3nosample
D3nosample^2+2*D3nosample+2
```

```
D3nosamplereflect<-(-(D3nosample^2+2*D3nosample+2))
D3nosamplereflect
yE.sample<-yE[idx==3]
yE.sample
yEnosample<-setdiff(yE,yE.sample)
yEnosample
TNWE3<-sum(yE.sample)+sum((3/4*((1-(M.sample-D3)/h1)^2))*yE.sample)
TNWE3
TRE3<-(sum(yE.sample)/sum(D3))*sum(mM3)
TRE3
Pi<-n1/N1
THTE3<-sum(yE.sample/n1*N1)
THTE3
TtrE3<-sum(yE.sample)+1/(n1*h1)*sum((3/4*(1-(Mnosample-D3nosamplereflect)/h1)^2)+(3/4*(1-
(Mnosample+D3nosamplereflect)/h1)^2)*yEnosample)
TtrE3

D4<-mM3[idx==4]
D4
mean(D4)
D4nosample<-setdiff(mM3,D4)
D4nosample
D4nosample^2+2*D4nosample+2
D4nosamplereflect<-(-(D4nosample^2+2*D4nosample+2))
D4nosamplereflect
yE.sample<-yE[idx==4]
yE.sample
yEnosample<-setdiff(yE,yE.sample)
yEnosample
TNWE4<-sum(yE.sample)+sum((3/4*((1-(M.sample-D4)/h1)^2))*yE.sample)
TNWE4
TRE4<-(sum(yE.sample)/sum(D4))*sum(mM3)
TRE4
Pi<-n1/N1
THTE4<-sum(yE.sample/n1*N1)
THTE4
TtrE4<-sum(yE.sample)+1/(n1*h1)*sum((3/4*(1-(Mnosample-D4nosamplereflect)/h1)^2)+(3/4*(1-
(Mnosample+D4nosamplereflect)/h1)^2)*yEnosample)
TtrE4
D5<-mM3[idx==5]
D5
mean(D5)
D5nosample<-setdiff(mM3,D5)
D5nosample
D5nosample^2+2*D5nosample+2
D5nosamplereflect<-(-(D5nosample^2+2*D5nosample+2))
D5nosamplereflect
yE.sample<-yE[idx==5]
yE.sample
yEnosample<-setdiff(yE,yE.sample)
yEnosample
TNWE5<-sum(yE.sample)+sum((3/4*((1-(M.sample-D5)/h1)^2))*yE.sample)
TNWE5
TRE5<-(sum(yE.sample)/sum(D5))*sum(mM3)
TRE5
```

```
Pi<-n1/N1
THTE5<-sum(yE.sample/n1*N1)
THTE5
TtrE5<-sum(yE.sample)+1/(n1*h1)*sum((3/4*(1-(Mnosample-D5nosamplereflect)/h1)^2)+(3/4*(1-
(Mnosample+D5nosamplereflect)/h1)^2)*yEnosample)
TtrE5
D6<-mM3[idx==6]
D6
mean(D6)
D6nosample<-setdiff(mM3,D6)
D6nosample
D6nosample^2+2*D6nosample+2
D6nosamplereflect<-(-(D6nosample^2+2*D6nosample+2))
D6nosamplereflect
yE.sample<-yE[idx==6]
yE.sample
yEnosample<-setdiff(yE,yE.sample)
yEnosample
TNWE6<-sum(yE.sample)+sum((3/4*((1-(M.sample-D6)/h1)^2))*yE.sample)
TNWE6
TRE6<-(sum(yE.sample)/sum(D6))*sum(mM3)
TRE6
Pi<-n1/N1
THTE6<-sum(yE.sample/n1*N1)
THTE6
TtrE6<-sum(yE.sample)+1/(n1*h1)*sum((3/4*(1-(Mnosample-D6nosamplereflect)/h1)^2)+(3/4*(1-
(Mnosample+D6nosamplereflect)/h1)^2)*yEnosample)
TtrE6

D7<-mM3[idx==7]
D7
mean(D7)
D7nosample<-setdiff(mM3,D7)
D7nosample
D7nosample^2+2*D7nosample+2
D7nosamplereflect<-(-(D7nosample^2+2*D7nosample+2))
D7nosamplereflect
yE.sample<-yE[idx==7]
yE.sample
yEnosample<-setdiff(yE,yE.sample)
yEnosample
TNWE7<-sum(yE.sample)+sum((3/4*((1-(M.sample-D7)/h1)^2))*yE.sample)
TNWE7
TRE7<-(sum(yE.sample)/sum(D7))*sum(mM3)
TRE7
Pi<-n1/N1
THTE7<-sum(yE.sample/n1*N1)
THTE7
TtrE7<-sum(yE.sample)+1/(n1*h1)*sum((3/4*(1-(Mnosample-D7nosamplereflect)/h1)^2)+(3/4*(1-
(Mnosample+D7nosamplereflect)/h1)^2)*yEnosample)
TtrE7
D8<-mM3[idx==8]
D8
mean(D8)
D8nosample<-setdiff(mM3,D8)
```

```
D8nosample
D8nosample^2+2*D8nosample+2
D8nosamplereflect<-(-(D8nosample^2+2*D8nosample+2))
D8nosamplereflect
yE.sample<-yE[idx==8]
yE.sample
yEnosample<-setdiff(yE,yE.sample)
yEnosample
TNWE8<-sum(yE.sample)+sum((3/4*((1-(M.sample-D8)/h1)^2))*yE.sample)
TNWE8
TRE8<-(sum(yE.sample)/sum(D8))*sum(mM3)
TRE8
Pi<-n1/N1
THTE8<-sum(yE.sample/n1*N1)
THTE8
TtrE8<-sum(yE.sample)+1/(n1*h1)*sum((3/4*(1-(Mnosample-D8nosamplereflect)/h1)^2)+(3/4*(1-
(Mnosample+D8nosamplereflect)/h1)^2)*yEnosample)
TtrE8
D9<-mM3[idx==9]
D9
mean(D9)
D9nosample<-setdiff(mM3,D9)
D9nosample
D9nosample^2+2*D9nosample+2
D9nosamplereflect<-(-(D9nosample^2+2*D9nosample+2))
D9nosamplereflect
yE.sample<-yE[idx==9]
yE.sample
yEnosample<-setdiff(yE,yE.sample)
yEnosample
TNWE9<-sum(yE.sample)+sum((3/4*((1-(M.sample-D9)/h1)^2))*yE.sample)
TNWE9
TRE9<-(sum(yE.sample)/sum(D9))*sum(mM3)
TRE9
Pi<-n1/N1
THTE9<-sum(yE.sample/n1*N1)
THTE9
TtrE9<-sum(yE.sample)+1/(n1*h1)*sum((3/4*(1-(Mnosample-D9nosamplereflect)/h1)^2)+(3/4*(1-
(Mnosample+D9nosamplereflect)/h1)^2)*yEnosample)
TtrE9
D10<-mM3[idx==10]
D10
mean(D10)
D10nosample<-setdiff(mM3,D10)
D10nosample
D10nosample^2+2*D10nosample+2
D10nosamplereflect<-(-(D10nosample^2+2*D10nosample+2))
D10nosamplereflect
yE.sample<-yE[idx==10]
yE.sample
yEnosample<-setdiff(yE,yE.sample)
yEnosample
TNWE10<-sum(yE.sample)+sum((3/4*((1-(M.sample-D10)/h1)^2))*yE.sample)
TNWE10
TRE10<-(sum(yE.sample)/sum(D10))*sum(mM3)
```

```
TRE10
Pi<-n1/N1
THTE10<-sum(yE.sample/n1*N1)
THTE10
TtrE10<-sum(yE.sample)+1/(n1*h1)*sum((3/4*(1-(Mnosample-D10nosamplereflect)/h1)^2)+(3/4*(1-
(Mnosample+D10nosamplereflect)/h1)^2)*yEnosample)
TtrE10

D11<-mM3[idx==11]
D11
mean(D11)
D11nosample<-setdiff(mM3,D11)
D11nosample
D11nosample^2+2*D11nosample+2
D11nosamplereflect<-(-(D11nosample^2+2*D11nosample+2))
D11nosamplereflect
yE.sample<-yE[idx==11]
yE.sample
yEnosample<-setdiff(yE,yE.sample)
yEnosample
TNWE11<-sum(yE.sample)+sum((3/4*((1-(M.sample-D11)/h1)^2))*yE.sample)
TNWE11
TRE11<-(sum(yE.sample)/sum(D11))*sum(mM3)
TRE11
Pi<-n1/N1
THTE11<-sum(yE.sample/n1*N1)
THTE11
TtrE11<-sum(yE.sample)+1/(n1*h1)*sum((3/4*(1-(Mnosample-D11nosamplereflect)/h1)^2)+(3/4*(1-
(Mnosample+D11nosamplereflect)/h1)^2)*yEnosample)
TtrE11
D12<-mM3[idx==12]
D12
mean(D12)
D12nosample<-setdiff(mM3,D12)
D12nosample
D12nosample^2+2*D12nosample+2
D12nosamplereflect<-(-(D12nosample^2+2*D12nosample+2))
D12nosamplereflect
yE.sample<-yE[idx==12]
yE.sample
yEnosample<-setdiff(yE,yE.sample)
yEnosample
TNWE12<-sum(yE.sample)+sum((3/4*((1-(M.sample-D12)/h1)^2))*yE.sample)
TNWE12
TRE12<-(sum(yE.sample)/sum(D12))*sum(mM3)
TRE12
Pi<-n1/N1
THTE12<-sum(yE.sample/n1*N1)
THTE12
TtrE12<-sum(yE.sample)+1/(n1*h1)*sum((3/4*(1-(Mnosample-D12nosamplereflect)/h1)^2)+(3/4*(1-
(Mnosample+D12nosamplereflect)/h1)^2)*yEnosample)
TtrE12
D13<-mM3[idx==13]
D13
mean(D13)
```

```
D13nosample<-setdiff(mM3,D13)
D13nosample
D13nosample^2+2*D13nosample+2
D13nosamplereflect<-(-(D13nosample^2+2*D13nosample+2))
D13nosamplereflect
yE.sample<-yE[idx==13]
yE.sample
yEnosample<-setdiff(yE,yE.sample)
yEnosample
TNWE13<-sum(yE.sample)+sum((3/4*((1-(M.sample-D13)/h1)^2))*yE.sample)
TNWE13
TRE13<-(sum(yE.sample)/sum(D13))*sum(mM3)
TRE13
Pi<-n1/N1
THTE13<-sum(yE.sample/n1*N1)
THTE13
TtrE13<-sum(yE.sample)+1/(n1*h1)*sum((3/4*(1-(Mnosample-D13nosamplereflect)/h1)^2)+(3/4*(1-
(Mnosample+D13nosamplereflect)/h1)^2)*yEnosample)
TtrE13
D14<-mM3[idx==14]
D14
mean(D14)
D14nosample<-setdiff(mM3,D14)
D14nosample
D14nosample^2+2*D14nosample+2
D14nosamplereflect<-(-(D14nosample^2+2*D14nosample+2))
D14nosamplereflect
yE.sample<-yE[idx==14]
yE.sample
yEnosample<-setdiff(yE,yE.sample)
yEnosample
TNWE14<-sum(yE.sample)+sum((3/4*((1-(M.sample-D14)/h1)^2))*yE.sample)
TNWE14
TRE14<-(sum(yE.sample)/sum(D14))*sum(mM3)
TRE14
Pi<-n1/N1
THTE14<-sum(yE.sample/n1*N1)
THTE14
TtrE14<-sum(yE.sample)+1/(n1*h1)*sum((3/4*(1-(Mnosample-D14nosamplereflect)/h1)^2)+(3/4*(1-
(Mnosample+D14nosamplereflect)/h1)^2)*yEnosample)
TtrE14
D15<-mM3[idx==15]
D15
mean(D15)
D15nosample<-setdiff(mM3,D15)
D15nosample
D15nosample^2+2*D15nosample+2
D15nosamplereflect<-(-(D15nosample^2+2*D15nosample+2))
D15nosamplereflect
sindex<-sample(300,20)
yE.sample<-yE[idx==15]
yE.sample
yEnosample<-setdiff(yE,yE.sample)
yEnosample
TNWE15<-sum(yE.sample)+sum((3/4*((1-(M.sample-D15)/h1)^2))*yE.sample)
```

```
TNWE15
TRE15<-(sum(yE.sample)/sum(D15))*sum(mM3)
TRE15
Pi<-n1/N1
THTE15<-sum(yE.sample/n1*N1)
THTE15
TtrE15<-sum(yE.sample)+1/(n1*h1)*sum((3/4*(1-(Mnosample-D15nosamplereflect)/h1)^2)+(3/4*(1-
(Mnosample+D15nosamplereflect)/h1)^2)*yEnosample)
TtrE15
conditional_bias1=c(13.48,15.65,15.10,13.51,14.47,15.02,12.32,12.99,13.33,12.10,13.93,15.14,13.39,1
2.65,15.60)
length(conditional_bias1)
conditional_bias2=c(2.99,3.70,4.56,-0.5,-0.72,-0.09,6.56,4.62,1.28,0.86,3.89,-0.64,1.14,0.006,8.23)
length(conditional_bias2)
conditional_bias3=c(2.02,5.54,4.73,-0.93,1.89,1.74,4.54,4.35,2.27,3.76,4.02,2.43,2.27,3.88,9.43)
length(conditional_bias3)
conditional_bias4=c(5.79,4.72,4.46,4.13,4.49,4.83,4.40,4.37,4.6,3.90,4.69,4.40,4.27,4.06,4.56)
length(conditional_bias4)
xbar.bar=c(0.7971,0.8233,0.8524,0.8565,0.9056,0.9479,0.9693,0.9711,0.9716,1.005,1.0287,1.0352,1.1
214,1.1413,1.3012)
length(xbar.bar)
plot(xbar.bar,conditional_bias1,type="l",col="1",lty=1,ylim=c(0,30),xlab="X.BAR.BAR",ylab="CON
DITIONAL BIAS",main="LINEAR FUNCTION")
lines(xbar.bar,conditional_bias2,type="l",col="2",lty=2)
lines(xbar.bar,conditional_bias3,type="l",col="3",lty=3)
lines(xbar.bar,conditional_bias4,type="l",col="4",lty=4)
legend(1.1,30,c("NADARAYA","RATIO","HORVITZ","TTR"),col=c(1,2,3,4),lty=c(1,2,3,4))

conditional_bias1=c(16.02,16.37,17.96,17.24,16.53,16.50,16.89,16.42,17.90,18.76,16.63,18.27,16.96,1
7.99,17.01)
length(conditional_bias1)
conditional_bias2=c(5.17,3.74,2.98,1.56,0.09,2.75,-0.96,1.66,3.59,7.61,2.20,4.89,-0.88,2.77,0.35)
length(conditional_bias2)
conditional_bias3=c(5.32,-0.25,2.75,0.45,-0.54,3.07,-0.56,2.45,3.66,7.71,2.39,4.26,-0.53,2.80,0.35)
length(conditional_bias3)
conditional_bias4=c(-0.5,-0.57,-0.78,-0.55,-0.6,-0.65,-0.79,-0.79,-0.86,-1.05,-0.68,-1.03,-0.74,-0.89,-
0.66)
length(conditional_bias4)
xbar.bar=c(1.0839,1.1205,1.1387,1.1642,1.1675,1.1707,1.1792,1.1815,1.1867,1.1874,1.1909,1.1937,1.
1986,1.2092,1.2176)
length(xbar.bar)
plot(xbar.bar,conditional_bias1,type="l",col="1",lty=1,ylim=c(0,30),xlab="X.BAR.BAR",ylab="CON
DITIONAL BIAS",main="QUADRATIC FUNCTION")
lines(xbar.bar,conditional_bias2,type="l",col="2",lty=2)
lines(xbar.bar,conditional_bias3,type="l",col="3",lty=3)
lines(xbar.bar,conditional_bias4,type="l",col="4",lty=4)
legend(1.18,30,c("NADARAYA","RATIO","HORVITZ","TTR"),col=c(1,2,3,4),lty=c(1,2,3,4))
conditional_bias1=c(1.28,1.56,1.52,1.94,1.22,0.89,1.16,1.96,0.57,1.63,1.16,1.10,1.32,0.79,1.84)
length(conditional_bias1)
conditional_bias2=c(0.005,3.79,1.83,10.39,1.3,10.71,1.01,6.49,5.85,2.29,3.97,2.15,0.28,3.11,4.35)
length(conditional_bias2)
conditional_bias3=c(0.48,3.59,1.86,4.32,1.05,5.10,1.39,6.65,7.17,2.02,2.76,2.36,0.21,4.19,6.50)
length(conditional_bias3)
conditional_bias4=c(-0.03,0.31,0.19,0.26,-0.06,0.14,-0.06,0.50,0.14,0.16,-0.03,-0.05,-0.01,-0.03,0.46)
length(conditional_bias4)
```

```
xbar.bar=c(0.0422,0.0743,0.0852,0.1064,0.1232,0.1241,0.1304,0.1396,0.1426,0.1520,0.1540,0.1587,0.
1654,0.1911,0.2392)
length(xbar.bar)
plot(xbar.bar,conditional_bias1,type="l",col="1",lty=1,ylim=c(0,30),xlab="X.BAR.BAR",ylab="CON
DITIONAL BIAS",main="EXPONENTIAL FUNCTION")
lines(xbar.bar,conditional_bias2,type="l",col="2",lty=2)
lines(xbar.bar,conditional_bias3,type="l",col="3",lty=3)
lines(xbar.bar,conditional_bias4,type="l",col="4",lty=4)
legend(0.15,30,c("NADARAYA","RATIO","HORVITZ","TTR"),col=c(1,2,3,4),lty=c(1,2,3,4))
```

APPENDIX 2: PUBLICATION