

## ABSTRACT

FERGUSON, BRADLEY THOMAS. Auxiliary Bootstrap Methods. (Under the direction of Eric Laber and Len Stefanski.)

The bootstrap has become a standard method for estimating characteristics of the sampling distribution of a functional. It provides approximations of quantiles, means, and variances that in some cases are more accurate than their corresponding counterparts. One of the main benefits of the bootstrap is its simplicity. By using Monte-Carlo simulation methods, it bypasses the need to derive complex asymptotic approximations by resampling the data a large number of times. One limitation of the bootstrap is its computational cost. If the functional of interest is computationally expensive, bootstrapping it a large number of times may not be feasible and thus a smaller number of resamples must be used, potentially leading to poorer approximations. We propose computational bootstrap methods which utilize a cheaper, correlated statistic in combination with the original statistic to construct what we call ‘auxiliary’ estimators of the mean, standard error, and quantile. These estimators rely on a strong relationship between the functional and corresponding surrogate and can be obtained at a fraction of the cost while maintaining a desired level of precision.

© Copyright 2018 by Bradley Thomas Ferguson

All Rights Reserved

Auxiliary Bootstrap Methods

by  
Bradley Thomas Ferguson

A dissertation submitted to the Graduate Faculty of  
North Carolina State University  
in partial fulfillment of the  
requirements for the Degree of  
Doctor of Philosophy

Statistics

Raleigh, North Carolina

2018

APPROVED BY:

---

Eric Laber  
Co-chair of Advisory Committee

---

Len Stefanski  
Co-chair of Advisory Committee

---

Brian Reich

---

Dennis Boos

---

James Bartlett

## DEDICATION

I dedicate my dissertation work to my parents, my children, and my wife.

At a young age my parents instilled values of hard work and constantly told me to pursue my goals. My mother in particular always told me that I could achieve whatever I wanted to in life. She was my greatest cheerleader growing up and I am forever grateful for this.

I dedicate this to my children, Thomas, Kate, and Lucy. They remind me everyday how beautiful life can be and have been a constant source of joy throughout this research process.

Lastly, I dedicate this to my wife Emily. She is my eternal companion and brings me more joy than words can describe. For all the late nights and work-filled weekends, she was there with me, believing in me every step of the way. I love her and could not have done this without her.

## ACKNOWLEDGEMENTS

I wish to thank my committee members who were more than generous with their expertise and precious time. A special thanks to Dr. Eric Laber and Dr. Len Stefanski for their countless hours of reflecting, reading, encouraging, and most of all patience throughout the entire process. Thank you Dr. Brian Reich, Dr. Dennis Boos, and Dr. James Bartlett for agreeing to serve on my committee.

I would like to acknowledge and thank the North Carolina State Statistics Department for allowing me to conduct my research and providing any assistance requested. Special thanks goes to the instructors I had who helped make statistics accessible and interesting.

I would also like to acknowledge my managers at Quintiles, Dr. Valerii Fedorov and Dr. Russell Reeves who were generous in giving me time off to work on parts of my dissertation. The same goes for my current manager at DOMO, Chris Error, who has been completely supportive of finishing my PhD. I must also acknowledge Carter Rees, my main co-worker at DOMO who provided assistance with performing some simulations on Amazon AWS.

Additionally I must also acknowledge the faculty in the Brigham Young University Statistics Department, particularly Dr. Natalie Blades, Dr. Shane Reese, and Dr. Gilbert Fellingham. They provided crucial guidance early on in my career and encouraged me to pursue a PhD. I would be remiss if I didn't also acknowledge Peter Dotson who provided invaluable computational resources that I was able to use in the final stages of my dissertation.

Finally I would like to thank Alan and Tonia Izu, my neighbors down the street from my childhood home who originally introduced statistics to me in high school and have been a constant source of information and support.

# TABLE OF CONTENTS

<b>LIST OF TABLES</b> . . . . .	<b>vi</b>
<b>LIST OF FIGURES</b> . . . . .	<b>vii</b>
<b>Chapter 1 Introduction</b> . . . . .	<b>1</b>
1.1 Introduction . . . . .	1
<b>Chapter 2 Review of Auxiliary Quantile Estimators</b> . . . . .	<b>3</b>
2.1 Introduction . . . . .	3
2.2 Methods . . . . .	4
2.2.1 Design-based estimators . . . . .	5
2.2.2 Model-based estimators . . . . .	9
2.2.3 Model-assisted estimators . . . . .	11
2.3 Simulation study . . . . .	18
2.3.1 Results . . . . .	20
2.4 Conclusion . . . . .	32
<b>Chapter 3 R Package for Auxiliary Quantile Estimation</b> . . . . .	<b>36</b>
3.1 Introduction . . . . .	36
3.2 Auxiliary quantile estimators . . . . .	37
3.2.1 Design-based estimators . . . . .	38
3.2.2 Model-based estimators . . . . .	39
3.2.3 Model-assisted generalized difference estimators . . . . .	40
3.2.4 Model-assisted calibration estimators . . . . .	41
3.3 Using the package . . . . .	43
3.3.1 Preparing the data . . . . .	43
3.3.2 Sample code for evaluating auxiliary estimators . . . . .	43
3.4 Conclusion . . . . .	47
<b>Chapter 4 Auxiliary Bootstrap Methods</b> . . . . .	<b>48</b>
4.1 Introduction . . . . .	48
4.2 Auxiliary bootstrap . . . . .	48
4.2.1 Auxiliary quantile estimator . . . . .	49
4.2.2 Auxiliary mean estimator . . . . .	50
4.2.3 Auxiliary variance estimator . . . . .	50
4.2.4 Choosing the number of surrogates . . . . .	51
4.2.5 Choosing which surrogates to label . . . . .	53
4.3 Proof of concept simulations . . . . .	54
4.3.1 Model . . . . .	54
4.3.2 Simulation design . . . . .	54
4.3.3 Results . . . . .	54
4.4 Simulation experiments . . . . .	55
4.4.1 Bayesian regression model . . . . .	58

4.4.2	Geospatial weather model . . . . .	60
4.5	Discussion . . . . .	62
<b>Chapter 5</b>	<b>Conclusion . . . . .</b>	<b>64</b>
<b>References</b>	<b>. . . . .</b>	<b>66</b>
<b>Appendix</b>	<b>. . . . .</b>	<b>71</b>
Appendix A	Additional Results . . . . .	72
A.1	Additional results from the proof of concept simulation experiment . . . . .	72

## LIST OF TABLES

Table 2.1	Two-way classification of the events $y \leq Q_y(\alpha)$ and $x \leq Q_x(\alpha)$ in the population.	9
Table 2.2	MASE values ( $\times 10^2$ ) and corresponding standard deviations for the design-based estimators with $n = 100$ and $\sigma = 0.1$ . The smallest MASE values for each generative model are <b>bolded</b> .	22
Table 2.3	MASE values ( $\times 10^2$ ) and corresponding standard deviations for the model-based estimators with $n = 100$ and $\sigma = 0.1$ . The smallest MASE values for each generative model are <b>bolded</b> .	24
Table 2.4	MASE values ( $\times 10^2$ ) and corresponding standard deviations for the generalized difference estimators with $n = 100$ and $\sigma = 0.1$ . The smallest MASE values for each generative model are <b>bolded</b> .	25
Table 2.5	MASE values ( $\times 10^2$ ) and corresponding standard deviations for the calibration estimators with $n = 100$ and $\sigma = 0.1$ . The smallest MASE values for each generative model are <b>bolded</b> .	29
Table 3.1	Description of variables needed for <b>data</b> .	43
Table 3.2	Description of variables needed for <b>GetModelCD_Q</b> .	45
Table 3.3	Description of variables needed for <b>GetModelK_Q</b> .	46

## LIST OF FIGURES

Figure 2.1	Finite populations of size $N = 2000$ generated from models 1-4 with $\sigma = 0.1$ (left) and $\sigma = 0.3$ (right). . . . .	21
Figure 2.2	Boxplots of ASE values ( $\times 10^2$ ) for the design-based estimators from models 1-4 with $n = 100$ and $\sigma = 0.1$ under SRS. . . . .	22
Figure 2.3	Boxplots of ASE values ( $\times 10^2$ ) for the design-based estimators from models 1-4 with $n = 100$ and $\sigma = 0.1$ under Poisson sampling. . . . .	23
Figure 2.4	Boxplots of ASE values ( $\times 10^2$ ) for the model-based estimators from models 1-4 with $n = 100$ and $\sigma = 0.1$ under SRS. . . . .	25
Figure 2.5	Boxplots of ASE values ( $\times 10^2$ ) for the model-based estimators from models 1-4 with $n = 100$ and $\sigma = 0.1$ under Poisson sampling. . . . .	26
Figure 2.6	Boxplots of ASE values ( $\times 10^2$ ) for the generalized difference estimators from models 1-4 with $n = 100$ and $\sigma = 0.1$ under SRS. . . . .	27
Figure 2.7	Boxplots of ASE values ( $\times 10^2$ ) for the generalized difference estimators from models 1-4 with $n = 100$ and $\sigma = 0.1$ under Poisson. . . . .	28
Figure 2.8	Boxplots of ASE values ( $\times 10^2$ ) for the calibration estimators from models 1-4 with $n = 100$ and $\sigma = 0.1$ under SRS . . . . .	30
Figure 2.9	Boxplots of ASE values ( $\times 10^2$ ) for the calibration estimators from models 1-4 with $n = 100$ and $\sigma = 0.1$ under Poisson sampling. . . . .	31
Figure 2.10	Boxplots of ASE values ( $\times 10^2$ ) from models 1-4 with $n = 100$ and $\sigma = 0.1$ under SRS . . . . .	32
Figure 2.11	Boxplots of ASE values ( $\times 10^2$ ) from models 1-4 with $n = 100$ and $\sigma = 0.1$ under Poisson sampling . . . . .	33
Figure 2.12	Plots of MSE values across increasing values of $\alpha$ for $n = 100$ and $\sigma = 0.1$ under SRS. . . . .	34
Figure 3.1	Simulated data set. . . . .	43
Figure 4.1	Plots of the MSE of $(\hat{\lambda}^*)^{-1}(L)$ , $(\hat{\lambda}^{aux})^{-1}(L)$ , and $(\hat{\lambda}^{aux})^{-1}(\hat{L})$ for estimating quantiles at $\alpha \in \{.80, .9, 95, .99\}$ across increasing $\rho$ where $L = 500$ and $C_x = .05$ . Also included is the average computational cost, $\hat{L}$ , used when computing $(\hat{\lambda}^{aux})^{-1}(\hat{L})$ (dashed line). . . . .	56
Figure 4.2	Plot of the MSE of $(\hat{\lambda}^*)^{-1}(L)$ , $(\hat{\lambda}^{aux})^{-1}(L)$ , and $(\hat{\lambda}^{aux})^{-1}(\hat{L})$ for estimating $\mu$ across increasing $\rho$ where $L = 500$ and $C_x = .05$ . Also included is the average computational cost, $\hat{L}$ , used when computing $(\hat{\lambda}^{aux})^{-1}(\hat{L})$ (dashed line). . . . .	57
Figure 4.3	Plot of the MSE of $(\hat{\lambda}^*)^{-1}(L)$ , $(\hat{\lambda}^{aux})^{-1}(L)$ , and $(\hat{\lambda}^{aux})^{-1}(\hat{L})$ for estimating $\sigma^2$ across increasing $\rho$ where $L = 500$ and $C_x = .05$ . Also included is the average computational cost, $\hat{L}$ , used when computing $(\hat{\lambda}^{aux})^{-1}(\hat{L})$ (dashed line). . . . .	57

Figure 4.4	Plots of $F_n$ against $S_n$ computed on 250 Monte Carlo data sets. . . . .	59
Figure 4.5	Plots of the AMSE of $\left(\widehat{Q}_n^*\right)^{-1}(\alpha, L)$ , $\left(\widehat{Q}_n^{aux}\right)^{-1}(\alpha, L)$ , and $\left(\widehat{Q}_n^{aux}\right)^{-1}(\alpha, \widehat{L})$ for $\alpha \in \{.80, .95\}$ and increasing computational budgets. Also included is the average computational cost, $\widehat{L}$ , used when computing $\left(\widehat{Q}_n^{aux}\right)^{-1}(\alpha, \widehat{L})$ (dashed line). The plots on the left display results for when $F_n$ is computed assuming an informative prior while the plots on the right display results for the noninformative prior. . . . .	61
Figure 4.6	Plots which show the correlation between $F_n$ and $S_n$ (left) and the average computation time of $S_n^d$ (right) for increasing values of $d$ based on 100 Monte Carlo data sets. . . . .	62
Figure 4.7	Plots of the AMSE of $\left(\widehat{Q}_n^*\right)^{-1}(\alpha, L)$ , $\left(\widehat{Q}_n^{aux}\right)^{-1}(\alpha, L)$ , and $\left(\widehat{Q}_n^{aux}\right)^{-1}(\alpha, \widehat{L})$ for $\alpha \in \{.80, .95\}$ and increasing computational budgets. Also included is the average computational cost, $\widehat{L}$ , used when computing $\left(\widehat{Q}_n^{aux}\right)^{-1}(\alpha, \widehat{L})$ (dashed line). . . . .	63
Figure A.1	Plots of the MSE of $\left(\widehat{\lambda}^*\right)^{-1}(L)$ , $\left(\widehat{\lambda}^{aux}\right)^{-1}(L)$ , and $\left(\widehat{\lambda}^{aux}\right)^{-1}(\widehat{L})$ for estimating quantiles at $\alpha \in \{.80, .9, .95, .99\}$ across increasing $\rho$ where $L = 100$ and $C_x = .1$ . Also included is the average computational cost, $\widehat{L}$ , used when computing $\left(\widehat{\lambda}^{aux}\right)^{-1}(\widehat{L})$ (dashed line). . . . .	73
Figure A.2	Plots of the MSE of $\left(\widehat{\lambda}^*\right)^{-1}(L)$ , $\left(\widehat{\lambda}^{aux}\right)^{-1}(L)$ , and $\left(\widehat{\lambda}^{aux}\right)^{-1}(\widehat{L})$ for estimating quantiles at $\alpha \in \{.80, .9, .95, .99\}$ across increasing $\rho$ where $L = 500$ and $C_x = .1$ . Also included is the average computational cost, $\widehat{L}$ , used when computing $\left(\widehat{\lambda}^{aux}\right)^{-1}(\widehat{L})$ (dashed line). . . . .	74
Figure A.3	Plots of the MSE of $\left(\widehat{\lambda}^*\right)^{-1}(L)$ , $\left(\widehat{\lambda}^{aux}\right)^{-1}(L)$ , and $\left(\widehat{\lambda}^{aux}\right)^{-1}(\widehat{L})$ for estimating quantiles at $\alpha \in \{.80, .9, .95, .99\}$ across increasing $\rho$ where $L = 100$ and $C_x = .05$ . Also included is the average computational cost, $\widehat{L}$ , used when computing $\left(\widehat{\lambda}^{aux}\right)^{-1}(\widehat{L})$ (dashed line). . . . .	75
Figure A.4	Plot of the MSE of $\left(\widehat{\lambda}^*\right)^{-1}(L)$ , $\left(\widehat{\lambda}^{aux}\right)^{-1}(L)$ , and $\left(\widehat{\lambda}^{aux}\right)^{-1}(\widehat{L})$ for estimating $\mu$ across increasing $\rho$ where $L = 100$ and $C_x = .1$ . Also included is the average computational cost, $\widehat{L}$ , used when computing $\left(\widehat{\lambda}^{aux}\right)^{-1}(\widehat{L})$ (dashed line). . . . .	76
Figure A.5	Plot of the MSE of $\left(\widehat{\lambda}^*\right)^{-1}(L)$ , $\left(\widehat{\lambda}^{aux}\right)^{-1}(L)$ , and $\left(\widehat{\lambda}^{aux}\right)^{-1}(\widehat{L})$ for estimating $\mu$ across increasing $\rho$ where $L = 500$ and $C_x = .1$ . Also included is the average computational cost, $\widehat{L}$ , used when computing $\left(\widehat{\lambda}^{aux}\right)^{-1}(\widehat{L})$ (dashed line). . . . .	76

Figure A.6	Plot of the MSE of $(\hat{\lambda}^*)^{-1}(L)$ , $(\hat{\lambda}^{aux})^{-1}(L)$ , and $(\hat{\lambda}^{aux})^{-1}(\hat{L})$ for estimating $\mu$ across increasing $\rho$ where $L = 100$ and $C_x = .05$ . Also included is the average computational cost, $\hat{L}$ , used when computing $(\hat{\lambda}^{aux})^{-1}(\hat{L})$ (dashed line).	77
Figure A.7	Plot of the MSE of $(\hat{\lambda}^*)^{-1}(L)$ , $(\hat{\lambda}^{aux})^{-1}(L)$ , and $(\hat{\lambda}^{aux})^{-1}(\hat{L})$ for estimating $\sigma^2$ across increasing $\rho$ where $L = 100$ and $C_x = .1$ . Also included is the average computational cost, $\hat{L}$ , used when computing $(\hat{\lambda}^{aux})^{-1}(\hat{L})$ (dashed line).	77
Figure A.8	Plot of the MSE of $(\hat{\lambda}^*)^{-1}(L)$ , $(\hat{\lambda}^{aux})^{-1}(L)$ , and $(\hat{\lambda}^{aux})^{-1}(\hat{L})$ for estimating $\sigma^2$ across increasing $\rho$ where $L = 500$ and $C_x = .1$ . Also included is the average computational cost, $\hat{L}$ , used when computing $(\hat{\lambda}^{aux})^{-1}(\hat{L})$ (dashed line).	78
Figure A.9	Plot of the MSE of $(\hat{\lambda}^*)^{-1}(L)$ , $(\hat{\lambda}^{aux})^{-1}(L)$ , and $(\hat{\lambda}^{aux})^{-1}(\hat{L})$ for estimating $\sigma^2$ across increasing $\rho$ where $L = 100$ and $C_x = .05$ . Also included is the average computational cost, $\hat{L}$ , used when computing $(\hat{\lambda}^{aux})^{-1}(\hat{L})$ (dashed line).	78

# Chapter 1

## Introduction

### 1.1 Introduction

Statistical methodology is increasingly focused on the development of flexible models for large and/or complex systems and consequently is becoming more computationally intensive [17]. The development and evaluation of modern statistical methods often relies heavily on Monte Carlo studies which may require fitting a statistical model tens or hundreds of thousands of times; thus, even when it is feasible to fit a model to a single dataset, it may not be possible to thoroughly examine its operating characteristics using standard methods of simulation experiments. Similarly, using the bootstrap or other resampling methods to estimate a functional of an estimator's sampling distribution may be prohibitively expensive. Given limited, i.e., finite, computational resources, researchers are often forced to choose between allocating clock-cycles to improve the quality of an estimator being fit to the dataset of interest and allocating clock-cycles to construct measures of uncertainty and/or evaluating the properties of the estimator using Monte Carlo methods.

In many settings, one can construct a computationally inexpensive surrogate for the estimator of interest which can be efficiently bootstrapped or evaluated in a simulation study. Canonical examples include emulators as surrogates for high-fidelity physical models [24, 48, 5], method of moments or maximum likelihood estimator as a surrogate for a posterior mode [7], and heuristic solutions as a surrogate for a global optimizer in M-estimation [6, 43]. A natural approach, when such a surrogate is available, is to use the surrogate to augment estimation of the sampling distribution of the estimator of interest. We propose a simple bootstrap procedure, which we term the auxiliary bootstrap, that: (i) computes the surrogate estimator on a large number of bootstrap data sets and the estimator of interest on a small subset of these; and (ii) uses classical methods from survey sampling with auxiliary data to approximate a functional of the sampling distribution of the estimator of interest. We consider an adaptive experimental

design for choosing estimator-surrogate pairs. Simulation examples suggest that the proposed methods can drastically reduce computation time without degrading solution quality even when the surrogate is systematically biased or only modestly correlated with the estimator of interest. While our developments are focused on the nonparametric bootstrap, the proposed methodology applies directly to other resampling schemes including the parametric bootstrap, jackknife, subsampling, and Monte Carlo studies in which data sets are drawn from the true generative model.

The problem of combining a high-quality but expensive estimator with a less expensive but potentially lower quality estimator has been studied extensively in the context of survey sampling with auxiliary information. These methods have not been widely used in the context of reducing the computational cost of Monte Carlo methods. One exception is [6] wherein a surrogate estimator was used to augment a jackknife estimator of the standard error for M-estimators; their jackknife estimator is a member of the class of estimators proposed here.

In Chapter 2, we conduct an extensive review of quantile estimators which leverage auxiliary information and perform a simulation experiment to compare each of the estimators. In Chapter 3, we introduce an R package `auxQuantile` which allows a user to implement the methods in Chapter 2. In Chapter 4 we formally introduce the auxiliary bootstrap framework. We perform a proof of concept simulation study on auxiliary bootstrap estimators of the mean, variance, and quantiles of a distribution. We then conduct two simulation studies on actual estimators and show that the auxiliary bootstrap estimators are 1) more precise under a constrained computational budget and 2) equally as precise as standard bootstrap methods but for a fraction of the cost. Lastly, we provide closing remarks in Chapter 5.

## Chapter 2

# Review of Auxiliary Quantile Estimators

### 2.1 Introduction

Technological advances and investment in ‘big-data’ infrastructure have dramatically increased the amount of data being collected and curated for statistical analyses. Much of this data collection is event-driven, i.e., data exhaust or transaction data, and is therefore subject to potential sampling bias or other pitfalls associated with non-experimental data [4, 16, 32, 25, 50, 26]. One approach to improve predictive and inferential statistical models built from large, observational data is to calibrate them using supplemental data collected in a carefully designed experiment. For example, the poor performance of the infamous Google Flu Trends model [28], built exclusively from search query data, was shown to improve substantially when supplemented with lagged data from the centers for disease control and prevention [36]. However, such supplemental data are potentially costly or burdensome to collect and therefore may be orders of magnitudes smaller than the observational data. Thus, there is growing interest in combining large, inexpensive, potentially low-quality data with small, expensive, high-quality data to construct high-quality statistical models. This problem has been studied extensively for decades in the survey sampling literature under the heading of estimation with auxiliary information [13, 22, 38]. The importance and utility of combining observational data and probability samples has been recently highlighted by [39].

However, despite a mature and vast literature on estimation with auxiliary information, an up-to-date review of these techniques accessible to non-sampling experts is lacking. Our purpose is to provide a systematic review of quantile estimation under auxiliary information that assumes no prior exposure to these estimators, thus making it accessible to the burgeoning cohort of data scientists with little or no training in sampling theory. We focus on estimation of quantiles under

auxiliary information as many functionals of interest can be expressed in terms of quantiles of the generative distribution. Furthermore, estimation of means and counts are well-documented in introductory sampling texts [12, 31]. In addition to this review, we present an extensive suite of simulation experiments to compare the performance of the reviewed methods. To our knowledge, this is the largest and most complete simulation study of quantile estimation under auxiliary information to date.

Our review covers 19 estimators that can be broadly grouped as design-based, model-based, or model-assisted. These estimators are catalogued in Section 2. We present a simulation study in Section 3 and concluding remarks in Section 4. Because of its popularity and utility we focus on quantile estimation and study a broad array of estimators. For a more in-depth study of a smaller set of distribution function estimators, see [45].

## 2.2 Methods

We assume a finite population indexed by  $U = \{1, \dots, N\}$ . Each member of the population has a scalar variable,  $y$ , which is of primary interest but potentially expensive to collect, and a scalar variable,  $x$ , which is inexpensive to collect and covaries with  $y$ . Let  $F_y(t) = N^{-1} \sum_{i \in U} I(y_i \leq t)$  denote the cumulative distribution (CDF) of  $y$  where  $I(\cdot)$  is the indicator function. For fixed but arbitrary  $\alpha \in (0, 1)$ , we consider estimation of  $Q_y(\alpha) = \inf \{t : F_y(t) \geq \alpha\}$ , the  $\alpha$ -quantile of the distribution of  $y$ . To estimate  $Q_y(\alpha)$ , we assume that we have available a sample of  $y$  values of size  $n$  indexed by  $s \subseteq U$  and all  $x$  values in the population. We assume that the sample of  $y$  values is drawn according to known probability distribution  $p(\cdot)$  over all subsets of  $U$  that satisfies  $p(s') > 0$  for all subsets  $s'$  of  $U$  of size  $n$ .

Quantile estimators with auxiliary information can be broadly classified as either CDF-based or direct-quantile-based. In CDF-based estimation, an estimator of  $Q_y(\alpha)$  is formed by inverting an estimator of  $F_y(t)$  and auxiliary information is used in the estimation of this CDF. To ensure that this inversion is well-defined and produces a non-degenerate estimator of  $Q_y(\alpha)$ , we require that the estimator of  $F_y(t)$ , say  $\tilde{F}_y(t)$ , be a proper CDF; i.e., with probability one: (P1)  $\tilde{F}_y(t)$  is right continuous; (P2)  $\tilde{F}_y(t)$  is monotone nondecreasing; (P3)  $\lim_{t \rightarrow -\infty} \tilde{F}_y(t) = 0$ ; and (P4)  $\lim_{t \rightarrow \infty} \tilde{F}_y(t) = 1$ . Some of the CDF estimators that we describe do not guarantee that these properties hold, in such cases we assume that post-estimation corrections are applied to ensure they are proper CDFs before inversion. For example, if (P2) does not hold we replace  $\tilde{F}_y(t)$  with  $\tilde{F}_y^*(t)$ , where  $\tilde{F}_y^* \{y_{(1)}\} = \tilde{F}_y \{y_{(1)}\}$  and  $\tilde{F}_y^* \{y_{(i)}\} = \max \left[ \tilde{F}_y^* \{y_{(i-1)}\}, \tilde{F}_y \{y_{(i)}\} \right]$ , where  $y_{(i)}$  is the  $i$ th order statistic of  $\{y_i\}_{i \in s}$  ([21]). If (P3) or (P4) do not hold then  $\tilde{F}_y(t)$  can be projected onto  $[0, 1]$ . All of the estimators considered here satisfy (P1).

The intuition driving CDF-based estimation is that a high-quality estimator of the CDF should produce a high-quality estimator of the target quantile. However, in CDF-based estima-

tion, auxiliary information is used to improve estimation of the entire CDF,  $F_y(t)$ , regardless of how close  $t$  is to  $Q_y(\alpha)$ . Direct-quantile-based estimators attempt to gain efficiency by using auxiliary information to improve estimation of  $F_y(t)$  in a neighborhood of  $t = Q_y(\alpha)$ .

We consider both design-based and model-based frameworks of inference. In a design-based framework all randomness arises from repeated sampling. Given a statistic  $\hat{\theta}(s)$  computed from the sample indexed by  $s \subseteq U$ , the sampling distribution of  $\hat{\theta}(s)$  is completely determined by the sampling mechanism,  $p(s)$ , used to generate  $s$ . The estimator  $\hat{\theta}(s)$  is said to be asymptotically design-unbiased if  $\mathbb{E}_p \left\{ \hat{\theta}(s) \right\} - \hat{\theta}(U) = \sum_{s' \subseteq U} \hat{\theta}(s') p(s') - \hat{\theta}(U)$  converges to zero as  $N \rightarrow \infty$ . In a model-based framework, the finite population  $\{(x_i, y_i)\}_{i \in U}$  are conceptualized as being a random draw from a fixed but unknown generative model, say  $\xi$ . Thus, the sampling distribution of a statistic, say  $\hat{\theta}(s)$ , constructed on a sample indexed by  $s$ , is dictated by  $\xi$ . The estimator  $\hat{\theta}(s)$  is said to be model-unbiased if  $\mathbb{E} \left\{ \hat{\theta}(s) - \hat{\theta}(U) \right\} = 0$  where the expectation is taken with respect to both the distribution of the finite population  $\{(x_i, y_i)\}_{i \in U}$ , governed by  $\xi$ , and the sampling mechanism, governed by  $p(s)$ .

We categorize estimators of  $Q_y(\alpha)$  as design-based, model-based, or model-assisted. Design-based estimators [14, 34, 33, 47, 46, 54, 58, 60] do not impose a superpopulation model for  $(y, x)$  and consequently are always asymptotically design-unbiased regardless of the underlying superpopulation model. However, design-unbiasedness is often achieved at the expense of increased variability. Furthermore, they are not guaranteed to produce proper CDFs and thus, as discussed previously, may need to be corrected before inversion.

Model-based estimators [8, 9, 14, 15, 30, 35, 40, 44] rely on a predictive model for  $y$  given  $x$ , and depend heavily on the superpopulation model assumptions. They are generally asymptotically model-unbiased and are efficient if the assumed model is correct, but need not be consistent if the postulated predictive model is incorrect. Thus, they are most useful in settings where adequate model checking is possible.

Model-assisted estimators [11, 23, 27, 42, 44, 47, 46, 51, 52, 53, 62, 63] are a hybrid of design- and model-based estimators. They rely on a superpopulation model to gain efficiency, but retain asymptotic design-unbiasedness if the model is misspecified.

### 2.2.1 Design-based estimators

The estimators surveyed in this section are asymptotically design-unbiased regardless of the underlying superpopulation model for  $(y, x)$ . Define  $\pi_i$  to be the probability of including the  $i$ th individual in the chosen sample under sampling design  $p(s)$  and let  $d_i = \pi_i^{-1}$ . If no auxiliary information is available, a common estimator of  $F_y(t)$  is the Horvitz-Thompson estimator [29]  $\hat{F}_{y,HT}(t) = N^{-1} \sum_{i \in s} d_i I(y_i \leq t)$ . Although this estimator is design-unbiased, it is not a proper distribution function if  $\sum_{i \in s} d_i \neq N$ . One way to ensure a proper distribution function

is to normalize the Horvitz-Thompson estimator by the inverse design weights  $\widehat{F}_{y,\text{HTN}}(t) = (\sum_{i \in s} d_i)^{-1} \sum_{i \in s} d_i I(y_i \leq t)$ . This estimator is asymptotically design-unbiased and more efficient than  $\widehat{F}_{y,\text{HT}}(t)$  (see Kuk, 1988 for details). It can be seen that  $\widehat{F}_{y,\text{HT}}(t) = \widehat{F}_{y,\text{HTN}}(t)$  for all  $t$  under simple random sampling (SRS) as  $\sum_{i \in s} d_i = N$ . Define  $\widehat{Q}_y(\alpha) = \inf \{t : \widehat{F}_{y,\text{HTN}}(t) \geq \alpha\}$  to be the quantile estimator based on inverting the normalized Horvitz-Thompson estimator. Similarly, let  $\widehat{F}_{x,\text{HTN}}(t)$  denote the normalized Horvitz-Thompson estimator for the CDF of  $x$  and  $\widehat{Q}_x(\alpha)$  the estimated quantile based on inverting  $\widehat{F}_{x,\text{HTN}}(t)$ .

### Ratio and difference estimators

Let  $\widehat{F}_y(t)$  and  $\widehat{F}_x(t)$  denote estimators of  $F_y(t)$  and  $F_x(t)$  (such as the Horvitz-Thompson estimators) computed from a sample  $s \subseteq U$ ; recall that  $F_x(t)$  is known because  $\{x_i\}_{i \in U}$  is observed. In the context of estimating means or totals with auxiliary information ratio and difference estimators are two of the most popular and well-studied approaches [12, 22]. The ratio estimator for a CDF is given by  $\widehat{F}_{y,\text{R|D}}(t) = \widehat{F}_y(t)F_x(t)/\widehat{F}_x(t)$  if  $\widehat{F}_x(t) \neq 0$ , and 0 otherwise. This estimator can be derived from an assumption of approximate proportionality  $F_y(t)/\widehat{F}_y(t) \approx F_x(t)/\widehat{F}_x(t)$  though the estimator can be justified more rigorously (see [12]). The difference estimator of the CDF is given by  $\widehat{F}_{y,\text{D|D}}(t) = \widehat{F}_y(t) + \{F_x(t) - \widehat{F}_x(t)\}$ . This estimator can be derived through an assumption of approximately equal differences,  $F_y(t) - \widehat{F}_y(t) \approx F_x(t) - \widehat{F}_x(t)$  (again, see [12] for additional discussion). For a description of the ratio and difference estimators as special cases within a broad class of estimators defined as smooth functions of  $\widehat{F}_y(t)$ ,  $\widehat{F}_x(t)$ , and  $F_x(t)$ , see [60]

Both the ratio and difference estimators are asymptotically design-unbiased [60] though they need not be monotone nor produce values in  $[0, 1]$  and thus are not proper CDFs. Let  $\widehat{Q}_{y,\text{D|R}}(\alpha)$  and  $\widehat{Q}_{y,\text{D|D}}(\alpha)$  denote the ratio and difference estimators of  $Q_y(\alpha)$  derived from suitably transformed  $\widehat{F}_{y,\text{D|R}}(t)$  and  $\widehat{F}_{y,\text{D|D}}(t)$ .

### Direct quantile-based ratio and difference estimators

An alternative to inverting the (suitably transformed) ratio and difference CDF estimators is to apply ratio and difference formulae to quantile estimators directly. Let  $\widehat{Q}_y(\alpha)$ ,  $\widehat{Q}_x(\alpha)$  denote estimators of  $Q_y(\alpha)$  and  $Q_x(\alpha)$ . Direct quantile-based ratio and difference estimators are  $\widehat{Q}_{y,\text{D|R}}^*(\alpha) = \widehat{Q}_y(\alpha)Q_x(\alpha)/\widehat{Q}_x(\alpha)$  for  $\widehat{Q}_x(\alpha) \neq 0$ , and  $\widehat{Q}_{y,\text{D|D}}^*(\alpha) = \widehat{Q}_y(\alpha) + \{Q_x(\alpha) - \widehat{Q}_x(\alpha)\}$  [55]. The ‘\*’ superscript is to distinguish these estimators from their CDF-based counterparts. Both  $\widehat{Q}_{y,\text{D|R}}^*(\alpha)$  and  $\widehat{Q}_{y,\text{D|D}}^*(\alpha)$  are asymptotically design-unbiased for  $Q_y(\alpha)$  and bypass the need to invert an improper CDF. Both  $\widehat{Q}_{y,\text{D|R}}^*(\alpha)$  and  $\widehat{Q}_{y,\text{D|D}}^*(\alpha)$  can be represented as special instances of a wider class of direct-quantile estimators of the form  $\widetilde{Q}(\alpha) = \widehat{Q}_y(\alpha) + \delta_0 \{Q_x(\alpha) - \widehat{Q}_x(\alpha)\}$  [54] where  $\delta_0 \geq 0$ .

## Post-stratification estimator

Another way to incorporate an auxiliary variable  $x$  into an estimator for  $F_y(t)$  is through post-stratification [58, 34]. Stratified sampling is often used in survey design to select a sample that is balanced across important characteristics of the population. In practice, it may not be possible to identify these characteristics or logistically feasible to collect a suitably stratified sample. In such cases, one can use auxiliary information to stratify and re-weight the sample after it is collected.

Suppose that  $U$  is partitioned into  $G$  poststrata  $U_1, U_2, \dots, U_G$  so that  $i \in U_g$  if and only if  $x_{(g-1)} < x_i \leq x_{(g)}$ , where  $-\infty = x_{(0)} < x_{(1)} < x_{(2)} < \dots < x_{(G-1)} < x_{(G)} = \infty$ . Let  $s_g = s \cap U_g$ ,  $N_g$  to be the size of  $U_g$ , and  $\hat{N}_g = \sum_{i \in s_g} \pi_i^{-1}$ ,  $g = 1, \dots, G$ . The post-stratification estimator is

$$\hat{F}_{y,D|PS}(t) = N^{-1} \sum_{g=1}^G \frac{N_g}{\hat{N}_g} \sum_{i \in s_g} d_i I(y_i \leq t) = \sum_{g=1}^G \tilde{\omega}_g \hat{F}_g(t), \quad (2.1)$$

where  $\tilde{\omega}_g = N^{-1}N_g$ , and  $\hat{F}_g(t)$  is the design-based estimator of  $F_g(t)$ , the CDF of  $y$  for the subpopulation  $U_g$ . Thus,  $\hat{F}_{D|PS}(t)$  is a weighted sum of design-based CDF estimators across the poststrata. Any poststrata with  $N_g = 0$  are pooled with adjacent poststrata until all  $N_g$  are positive. Three proposed choices of poststrata are:

- (E) equal poststrata size so that  $N_g \approx N/G$  for  $g = 1, \dots, G$ ;
- (T) equal aggregate size so that  $\sum_{i \in U_g} x_i \approx G^{-1} \sum_{i \in U} x_i$  for  $g = 1, \dots, G$ ;
- (R) equal aggregate square root size so that  $\sum_{i \in U_g} \sqrt{x_i} \approx G^{-1} \sum_{i \in U} \sqrt{x_i}$  for  $g = 1, \dots, G$ .

In the above expressions, one can choose strata by minimizing the discrepancy between the within strata and population level statistics, e.g., in (T) one might choose  $U_1, \dots, U_G$  so as to minimize  $\sum_{g=1}^G \left\{ \sum_{i \in U_g} x_i - G^{-1} \sum_{i \in U} x_i \right\}^2$ . [58] compared (E), (T), and (R) using simulation experiments under an SRS design and found that (E) with  $G = 4$  had a low probability of producing empty strata and led to small mean-squared error of  $\hat{F}_{y,D|PS}(t)$  relative to other choices for poststrata. Because  $\hat{F}_{D|PS}(t)$  is a convex combination of proper CDFs, it too is a proper CDF and can thus be inverted to obtain  $\hat{Q}_{y,D|PS}(\alpha)$ .

## Nonparametric density estimator

The estimators considered so far modify the estimated CDF or quantile of  $y$  using the CDF or quantile of auxiliary variable  $x$ . Thus, these adjustments are coarse in the sense that they do not use local information about the relationship between  $x$  and  $y$ . Writing  $F_y(t) = \int F_{y|x}(t|x) dF_x(x)$  suggests another strategy wherein one first constructs an estimator of the conditional CDF

$F_{y|x}(t|x)$ , say  $\widehat{F}_{y|x}(t|x)$ , and subsequently the estimator  $N^{-1} \sum_{i \in U} \widehat{F}_{y|x}(t; x_i)$ . An estimator of  $F_{y|x}(t|x)$  can be constructed using standard methods (e.g., see [57] and references therein). [33] proposed the following estimator

$$\widehat{F}_{y|x}(t; x) = \frac{\sum_{i \in s} d_i w\{b^{-1}(x - x_i)\} W\{b^{-1}(t - y_i)\}}{\sum_{i \in s} d_i w\{b^{-1}(x - x_i)\}},$$

where  $W(u) = \exp(u) / \{1 + \exp(u)\}$  is the logistic distribution function,  $w(u) = W'(u)$ , and  $b > 0$  is a tuning parameter. This estimator is derived from a bivariate density estimator for  $F_{x,y}(x, y)$  with the product of logistic kernels. Kuk proposed the ad hoc tuning rule  $b \approx \text{range}(x)/n$ , though other data-driven tuning procedures could be used [59, 57]. In the simulation experiments in Section 2.3, we set  $b = 1.06\hat{\sigma}_x n^{-1/5}$ , where  $\hat{\sigma}_x$  is the sample standard deviation of the  $x$  values, as proposed by [59].

Define  $\widehat{F}_{y, \text{D|DEN}}(t) = N^{-1} \sum_{i \in U} \widehat{F}_{y|x}(y|x_i)$ , where  $\widehat{F}_{y|x}(y|x)$  is as defined above. It can be seen that  $\widehat{F}_{y, \text{D|DEN}}(t)$  is a proper CDF. Let  $\widehat{Q}_{y, \text{D|DEN}}(\alpha)$  denote the quantile estimator obtained by inverting this estimator.

### Position estimator

The position estimator [34] seeks to increase efficiency by exploiting concordance, or agreement, between the events  $y \leq Q_y(\alpha)$  and  $x \leq Q_x(\alpha)$ ; an assumption of concordance between these events may be more tenable than an assumption of linearity between the quantile functions  $Q_y(\alpha)$  and  $Q_x(\alpha)$  as assumed by the ratio estimator. Let  $y_{(1)}, \dots, y_{(n)}$  denote the ordered elements of  $y$  indexed by  $s$ . Let  $i^* = \arg \max \{y_i : i \in s, y_{(i)} \leq Q_y(\alpha)\}$ , i.e.,  $i^*$  denotes the (unknown) position of  $Q_y(\alpha)$  in the observed sample. Define  $p^* = i^*/n$  and let  $\widehat{Q}_y(\cdot)$  denote an estimator of the quantile function. In finite samples,  $p^*$  need not equal  $\alpha$  and in such cases  $\widehat{Q}_y(p^*)$  may be a more efficient estimator of  $Q_y(\alpha)$  than  $\widehat{Q}_y(\alpha)$ . The position estimator uses auxiliary information to construct an estimator of  $p^*$ , say  $\widehat{p}$ , and subsequently the estimator  $\widehat{Q}_y(\widehat{p})$  of  $Q_y(\alpha)$ .

For simplicity we explain the position estimator under SRS. Table 2.1 shows the proportions of elements in  $U$  cross-classified by the events  $y \leq Q_y(\alpha)$  and  $x \leq Q_x(\alpha)$ . The proportions in the cells of this table are generally unknown but the column sums  $p_{\cdot j} = p_{1j} + p_{2j}$  for  $j = 1, 2$  are observed. For each  $i, j$ , define  $\widehat{p}_{ij}$  to be the plug-in estimator of  $p_{ij}$  based on sample  $s$ , i.e.,  $\widehat{p}_{11} = n^{-1} \sum_{i \in s} I\{y_i \leq \widehat{Q}_y(\alpha), x_i \leq \widehat{Q}_x(\alpha)\}$ , and let  $\widehat{p}_{\cdot j} = \widehat{p}_{1j} + \widehat{p}_{2j}$ . Define  $\widehat{p} = \widehat{p}_{11}(\widehat{p}_{\cdot 1}/p_{\cdot 1}) + \widehat{p}_{12}(\widehat{p}_{\cdot 2}/p_{\cdot 2})$  and subsequently  $\widehat{Q}_{y, \text{D|POS}}(\alpha) = \widehat{Q}_y(\widehat{p})$ . The estimator  $\widehat{p}$  of  $p^*$  can be viewed as a re-weighted version of the naive estimator  $\widehat{p}_{11} + \widehat{p}_{12}$  where the weights are based on ratios derived from the auxiliary variable  $x$ ; see Kuk and Mak (1989) for asymptotic properties of the position estimator.

Table 2.1: Two-way classification of the events  $y \leq Q_y(\alpha)$  and  $x \leq Q_x(\alpha)$  in the population.

	$x \leq Q_x(\alpha)$	$x > Q_x(\alpha)$
$y \leq Q_y(\alpha)$	$p_{11}$	$p_{12}$
$y > Q_y(\alpha)$	$p_{21}$	$p_{22}$

### 2.2.2 Model-based estimators

The estimators considered so far were motivated from a design-based approach to statistical inference. We now turn to cataloging estimators derived from a model-based perspective, i.e., where the finite population is treated as arising from a hypothesized superpopulation. The divide between model- and design-based inference dates back to Pearson and Fisher; for a historical account see [37].

Many common approaches to estimating a CDF using auxiliary information from a model-based perspective are based on the decomposition

$$F_y(t) = N^{-1} \left\{ \sum_{i \in s} I(y_i \leq t) + \sum_{i \in U \setminus s} I(y_i \leq t) \right\}.$$

In light of this decomposition, a natural approach to estimating  $F_y(t)$  is to construct an estimator, say  $\hat{g}(x, t)$ , of  $g(x, t) = P(y \leq t|x)$  using  $s$  and subsequently to construct the estimator

$$\hat{F}_M(t) = N^{-1} \left\{ \sum_{i \in s} I(y_i \leq t) + \sum_{i \in U \setminus s} \hat{g}(x_i, t) \right\}. \quad (2.2)$$

Thus, for  $i \in U \setminus s$ ,  $I(y_i \leq t)$  is replaced with an estimator of its conditional expectation constructed from  $s$ . All of the estimators considered in this section have this form and differ in the postulated model for  $g(x, t)$ .

It can be seen that if  $\hat{g}(x, t)$  is asymptotically unbiased for all  $t$  and all  $x_i, i \in U \setminus s$ , then  $\hat{F}_M(t)$  is asymptotically model-unbiased for  $F_y(t)$ . In addition, if  $\hat{g}(x_i, t)$  is strictly non-negative, monotone non-decreasing, with asymptotes at zero and one for  $i \in U \setminus s$ , then  $\hat{F}_M(t)$  is a proper CDF.

#### Chambers and Dunstan estimator

The Chambers and Dunstan estimator ([8]) postulates a location-scale model of the form  $y_i = \mu(x_i) + \sigma(x_i)\epsilon_i$  for  $i \in U$  where  $\mu(x)$  and  $\sigma(x)$  are smooth functions, and the errors  $\epsilon_1, \dots, \epsilon_N$  are i.i.d. with mean zero and unit variance. Let  $G$  denote the CDF of  $\epsilon$ , so that under this location-scale model  $g(x, t) = G[\{t - \mu(x)\}/\sigma(x)]$ . An estimator of  $g(x, t)$  is formed by constructing

estimators of: (i)  $\mu(x)$ , say  $\hat{\mu}(x)$ , using (non)linear least-squares; (ii)  $\sigma(x)$ , say  $\hat{\sigma}(x)$ , using variance modeling; and (iii)  $G(\epsilon)$ , say  $\hat{G}(\epsilon)$ , using the observed residuals  $\{y_i - \hat{\mu}(x_i)\} / \hat{\sigma}(x_i)$ . For example, in their original paper, Chambers and Dunstan postulate a linear working model  $\mu(x) = \beta_0 + \beta_1 x$  indexed by parameters  $\beta_0, \beta_1 \in \mathbb{R}$ , assume that  $\sigma(x)$  is known, and use the empirical distribution of the residuals to estimate  $G(\epsilon)$ . Let  $\hat{\beta}_0, \hat{\beta}_1$  denote the weighted least squares estimators of  $\beta_0, \beta_1$  with weight  $1/\sigma(x_i)$  for observation  $i \in s$ . Thus, the Chambers and Dunstan model is (2.2) with  $\hat{g}(x_i, t) = \hat{G} \left[ \left\{ t - \hat{\beta}_0 - \hat{\beta}_1 x_i \right\} / \sigma(x_i) \right]$ , where  $\hat{G}$  is the empirical CDF of the observed residuals; see [44] for an extension to non-identically distributed errors. Let  $\hat{Q}_{y,CD}(\alpha)$  denote the estimator of  $Q_y(\alpha)$  obtained by inverting this estimator of the CDF.

[40] proposed to use the Chambers and Dunstan estimator with a kernel-based estimator for  $\sigma(x)$ . Let  $k(\cdot)$  denote a kernel function,  $b$  a bandwidth, and  $\tilde{\beta}_0, \tilde{\beta}_1$  the unweighted least squares estimators of  $\beta_0, \beta_1$ . The proposed kernel-based estimator of  $\sigma(x)$  is

$$\hat{\sigma}^2(x) = \frac{\sum_{i \in s} k\{(x - x_i)/b\} (y_i - \tilde{\beta}_0 - \tilde{\beta}_1 x_i)^2}{\sum_{i \in s} k\{(x - x_i)/b\}}.$$

Parameters  $\beta_0, \beta_1$  indexing the mean function are then estimated using weighted least squares with weights  $1/\hat{\sigma}(x_i)$  for each  $i \in s$ ; let  $\hat{\beta}_{0,L}, \hat{\beta}_{1,L}$  denote these estimators. The Lombardia estimator of  $F_y(\alpha)$  is obtained using (2.2) with  $\hat{g}(x, t) = \hat{G} \left[ \left\{ t - \hat{\beta}_{0,L} - \hat{\beta}_{1,L} x \right\} / \hat{\sigma}(x) \right]$ , where  $\hat{G}$  is the empirical CDF of the observed residuals. Let  $\hat{Q}_{y,LGP}(\alpha)$  denote the estimator of  $Q_y(\alpha)$  obtained by inverting the preceding CDF estimator.

Additional flexibility can be added by allowing a nonlinear model for the mean function  $\mu(x)$ . [14] use a kernel-based estimator of the mean; [30] use local linear regression; and [35] uses  $k$ -nearest neighbors. As with all regression models, there is an inherent robustness-efficiency trade-off associated with modeling the mean, variance, and residual distribution in construction of Chambers and Dunstan type estimators. We explore this trade-off in simulation experiments in Section 2.3.

### Kuo estimator

In some settings, a location-scale model may be too restrictive. The [35] estimator is based on the more general model  $P(y \leq t|x) = g(x, t)$  where  $g(x, t)$  is an unknown smooth function of  $x$ . An estimator of  $g(x, t)$  is  $\hat{g}(x, t) = \sum_{j \in s} \omega_j(x) I(y_j \leq t)$  where  $\omega_j(x), j \in s$ , are weights satisfying  $\omega_j(x) \geq 0$  for  $j \in s$  and  $\sum_{j \in s} \omega_j(x) = 1$ . Kuo proposed the following candidate weights: (1) naive,  $\omega_j(x) = I(|x_j - x| < \delta) / \sum_{i \in s} I(|x_i - x| < \delta)$  for  $\delta > 0$ ; (2) Nadaraya-Watson,  $\omega_j(x) = K\{(x_j - x)/b\} / \sum_{i \in s} K\{(x_i - x)/b\}$  for some kernel function  $K(\cdot)$  and bandwidth  $b > 0$ ; and (3)  $k$ -nearest neighbor, where  $\omega_j(x) = k^{-1}$  if  $x_j$  is one of the  $k$  nearest neighbors to  $x$ , and 0 otherwise. We denote  $\hat{Q}_K(\alpha)$  to be the quantile estimator based on inverting this estimator

of the CDF.

### Chambers, Dunstan, and Wherly estimators

The estimator  $\widehat{Q}_K(\alpha)$  has the appeal of capturing a wide range of models between  $y$  and  $x$  but can be inefficient if a location-scale model is sufficient. The Chambers, Dunstan, and Wherly ([9]) estimator augments the Kuo estimator and gains efficiency when the location-scale model is correctly specified while maintaining (asymptotic) unbiasedness under a more general class of models. If the general model  $P(y \leq x|t) = g(x, t)$  is assumed when the location-scale model  $y_i = \mu(x_i) + \sigma(x_i)\epsilon_i$ ,  $i \in U$ , is true, then the expected bias in estimating  $P(y \leq t|x)$  with  $\hat{g}(x, t) = \sum_{i \in s} \omega_i(x)I(y_i \leq t)$  is

$$\delta(x, t) = \mathbb{E}[\hat{g}(x, t) - I(y \leq t)] = \sum_{i \in s} \omega_i(x)G\left(\frac{t - \mu(x_i)}{\sigma(x_i)}\right) - G\left(\frac{t - \mu(x)}{\sigma(x)}\right).$$

This is estimated by

$$\hat{\delta}(x, t) = \sum_{i \in s} \omega_i(x)\widehat{G}\left(\frac{t - \widehat{\mu}(x_i)}{\widehat{\sigma}(x_i)}\right) - \widehat{G}\left(\frac{t - \widehat{\mu}(x)}{\widehat{\sigma}(x)}\right),$$

where  $\widehat{\mu}(x)$ ,  $\widehat{\sigma}(x)$ , and  $\widehat{G}(t)$  are estimators of  $\mu(x)$ ,  $\sigma(x)$ , and  $G(t)$ . A bias-robust estimator of  $g(x, t)$  is  $\hat{g}(x, t) = \sum_{i \in s} \omega_i(x)I(y_i \leq t) - \hat{\delta}(x, t)$ . This estimator should be more efficient than  $\hat{g}(x, t) = \sum_{i \in s} \omega_i(x)I(y_i \leq t)$  when a location-scale model holds, even approximately, and should exhibit similar robustness when it does not hold [9].

However,  $\hat{g}(x, t)$  is not guaranteed to be monotone increasing or contained in  $[0, 1]$ , thus the corresponding CDF is not necessarily proper and may need to be transformed prior to inversion. We denote  $\widehat{Q}_{M|CDW}(\alpha)$  to be the quantile estimator based on this inversion.

### 2.2.3 Model-assisted estimators

Model-assisted estimators combine aspects of design and model-based estimators. Unlike model-based estimators, they account for sampling design and remain asymptotically design-unbiased despite a misspecified model. Unlike design-based estimators, they rely on a superpopulation model to gain efficiency if the assumed model is correctly specified. The two most common classes of model-assisted estimators are generalized difference estimators and model-calibrated estimators.

There is a conceptual difference between how models are fit in the model-based and model-assisted frameworks. To illustrate this difference, consider the standard linear model  $y_i = \mathbf{x}_i^T \boldsymbol{\theta} + \epsilon_i$ , where  $\epsilon_i$  are i.i.d. mean zero random variables. When a model-based approach is taken,

$(y_i, x_i)$ ,  $i \in s$ , are viewed as an i.i.d. sample from a superpopulation and  $\boldsymbol{\theta}$  is estimated using standard model-fitting procedures for i.i.d. data. Under the design-based paradigm, the finite population  $(y_i, x_i)$ ,  $i \in U$ , are not seen as samples from a superpopulation. In this case,  $\boldsymbol{\theta}_N$  replaces  $\boldsymbol{\theta}$  where  $\boldsymbol{\theta}_N$  is a fixed estimator of  $\boldsymbol{\theta}$  based on the full finite population. We define  $\widehat{\boldsymbol{\theta}}_N$  to be a design-based estimator of  $\boldsymbol{\theta}_N$  from the observed sample. If  $\boldsymbol{\theta}_N$  can be expressed as a function of population means or totals, then  $\widehat{\boldsymbol{\theta}}_N$  can be computed by replacing the means or totals with their design-weighted counterparts. For example, consider the standard least squares estimator,  $\boldsymbol{\theta}_N = (\mathbf{X}_N^T \mathbf{X}_N)^{-1} \mathbf{X}_N^T \mathbf{y}_N$ , where  $\mathbf{X}_N$  is a design matrix constructed from the entire finite population and  $\mathbf{y}_N$  is its corresponding vector of the finite population  $y$  values. Define  $\mathbf{D} = \text{diag}(d_1, \dots, d_n)$  to be a diagonal matrix of the sample design weights. A design-based estimator of  $\boldsymbol{\theta}_N$  is  $\widehat{\boldsymbol{\theta}}_N = (\mathbf{X}_n^T \mathbf{D} \mathbf{X}_n)^{-1} \mathbf{X}_n^T \mathbf{D} \mathbf{y}_n$  where  $\mathbf{X}_n$  and  $\mathbf{y}_n$  are as previously defined but for the observed sample.

The model-assisted estimators described in this section are functions of design-based estimators. For a more thorough discussion of estimating parameters in a model and design-based frameworks, see [23].

### Generalized difference estimators

Generalized difference estimators of  $F_y(t)$  have the form

$$\widehat{F}_{\text{GD}}(t) = N^{-1} \left[ \sum_{i \in s} d_i I(y_i \leq t) + \sum_{i \in U} \hat{g}(x_i, t) - \sum_{i \in s} d_i \hat{g}_c(x_i, t) \right], \quad (2.3)$$

where  $\hat{g}(x_i, t)$  is a design-based estimator of  $P(y_i \leq t | x_i)$  based on a superpopulation model, and  $\hat{g}_c(x_i, t)$  is a similar estimator conditional on  $i \in s$  (see Section 2.2.3 below). Each of the estimators we examine in this section differ by the choice of  $\hat{g}(x, t)$  and  $\hat{g}_c(x_i, t)$ . The idea of generalized difference estimators is to construct an estimator that is asymptotically model-unbiased under a correctly specified model while retaining asymptotic design-unbiasedness under an incorrectly specified model. These estimators are not always strictly monotone increasing nor bounded by 0 and 1, and thus may need to be transformed before inverting to obtain a quantile.

### Rao, Kovar, and Mantel estimator

The Rao, Kovar, and Mantel estimator ([47]) postulates a location-scale model as in [8]. An estimator of  $g(x, t)$  is calculated from design-based estimators of  $\mu(x)$ ,  $\sigma(x)$  and  $G(t)$  as:

$$\hat{g}(x_i, t) = \left( \sum_{j \in s} d_j \right)^{-1} \left\{ \sum_{j \in s} d_j I[\epsilon_j \leq \{t - \widehat{\mu}(x_i)\} / \widehat{\sigma}(x_i)] \right\},$$

and  $\hat{g}_c(x_i, t) = \left( \sum_{j \in s} d_{j|i} \right)^{-1} \left\{ \sum_{j \in s} d_{j|i} I[\epsilon_j \leq \{t - \hat{\mu}(x_i)\} / \hat{\sigma}(x_i)] \right\}$ , where  $\epsilon_j$  are the observed residuals and  $d_{i|j} = \pi_j / \pi_{ij}$  is the inverse of the conditional probability that  $i \in s$  given  $j \in s$ . The CDF based on these estimators is asymptotically model and design-unbiased. [63] proposed a similar generalized difference estimator with  $\hat{g}_c(x_i, t) = \hat{g}(x_i, t)$ ,  $i \in s$ . Let  $\hat{Q}_{y, \text{GD|RKM}}(\alpha)$  be the estimator of  $Q_y(\alpha)$  obtained after inverting the CDF estimator.

### Wu and Sitter estimator

The Rao, Kovar, and Mantel estimator does not necessarily have the property  $\hat{Q}_{y, \text{GD|RKM}}(\alpha) = Q_y(\alpha)$  if  $\mathbb{E}(y_i|x_i) = y_i$ ,  $i \in U$ , whereas the [63] estimator satisfies this property by constructing an estimator of  $I\{\mathbb{E}(y|x) \leq t\}$  instead of  $\mathbb{E}\{I(y \leq t)|x\}$ . The former requires postulating a model for  $\mathbb{E}(y|x) = \mu(x)$  which may be more straightforward than modeling  $\mathbb{E}\{I(y < t)|x\}$ . The CDF estimator simplifies to

$$\hat{F}_{y, \text{GD|WS}}(t) = \hat{F}_{y, \text{HT}}(t) + \left\{ F_{\hat{\mu}}(t) - \hat{F}_{\hat{\mu}}(t) \right\}, \quad (2.4)$$

where  $F_{\hat{\mu}}(t) = N^{-1} \sum_{i \in U} I\{\hat{\mu}(x_i) \leq t\}$  is the CDF of the predicted  $\hat{\mu}(x_i)$  values, and  $\hat{F}_{\hat{\mu}}(t) = N^{-1} \sum_{i \in s} d_i I\{\hat{\mu}(x_i) \leq t\}$  is the corresponding sample-based estimator. This estimator is not asymptotically model-unbiased but is design-consistent. Let  $\hat{Q}_{y, \text{GD|WS}}(\alpha)$  be the estimator of  $Q_y(\alpha)$  obtained by inverting the transformed CDF in (3.2).

### Chen and Wu direct quantile estimator

A similar estimator that relies on the same modeling assumptions but bypasses the need to transform an improper CDF is the direct-quantile generalized difference estimator [11],

$$\hat{Q}_{y, \text{GD|CW}}^*(\alpha) = \hat{Q}_y(\alpha) + Q_{\hat{\mu}}(\alpha) - \hat{Q}_{\hat{\mu}}(\alpha),$$

where  $Q_{\hat{\mu}}(\alpha)$  and  $\hat{Q}_{\hat{\mu}}(\alpha)$  are obtained by inverting  $F_{\hat{\mu}}(t)$ , and  $\hat{F}_{\hat{\mu}}(t)$ . The ‘\*’ is used to distinguish it from the CDF-based quantile estimators proposed by [11]. Because  $\hat{F}_{y, \text{HT}}(t)$ ,  $F_{\hat{\mu}}(t)$ , and  $\hat{F}_{\hat{\mu}}(t)$  are all proper CDFs,  $\hat{Q}_{y, \text{GD|CW}}^*(\alpha)$  is well-defined.

### Kuo type estimator

Similar to the model-based Kuo estimator, the generalized difference Kuo estimator [14, 30] is based on the less restrictive model  $P(y \leq t|x) = g(x, t)$ , where for each fixed  $t$ ,  $g(x, t)$  is an unknown smooth function of  $x$ . The estimator of  $g(x, t)$  is obtained using design-weighted nonparametric methods. For example, [30] use the design-weighted, local-linear regression estimator  $\hat{g}(x, t) = [1 \ 0](\mathbf{X}(x)^T \mathbf{W}(x) \mathbf{X}(x))^{-1} \mathbf{X}(x)^T \mathbf{W}(x) I(\mathbf{y} \leq t)$ , where  $[1 \ 0]$  is a  $1 \times 2$  vector,  $\mathbf{X}(x) = (1 \ x_i - x)_{i \in s}$ , and  $\mathbf{W}_x = \text{diag} [b^{-1} K \{b^{-1}(x_i - x)\} d_i]_{i \in s}$  for some kernel function  $K$

and bandwidth  $b > 0$ . This CDF estimator is asymptotically model and design-unbiased and we denote  $\widehat{Q}_{y, \text{GD|K}}(\alpha)$  to be the corresponding quantile estimator obtained after inverting the CDF.

### Hybrid estimator

The hybrid estimator [62] is a weighted sum of the [8] model-based estimator and the Rao, Kovar, and Mantel [47] estimator. Wang and Dorfman provide a derivation of the CDF under the assumed model  $y_i = \beta_0 + \beta_1 x_i + \epsilon_i$  and under simple random sampling. They define the CDF estimator  $\widehat{F}_H(t) = \phi_{opt} \widehat{F}_{M|CD}(t) + (1 - \phi_{opt}) \widehat{F}_{GD|CD}(t)$ , where  $\phi_{opt}$  is the value of  $\phi$  that minimizes the asymptotic mean square error of  $\widehat{F}_H(t) = \phi \widehat{F}_{M|CD}(t) + (1 - \phi) \widehat{F}_{GD|RKM}(t)$ . Under the assumed linear model and simple random sampling, this estimator is asymptotically model and design-unbiased and has a smaller asymptotic variance than either  $\widehat{F}_{M|CD}(t)$  or  $\widehat{F}_{GD|RKM}(t)$ . Because  $\widehat{F}_H(t)$  involves  $\widehat{F}_{GD|RKM}(t)$ , it is not a valid CDF and may need to be transformed to obtain a quantile estimator.

### Model calibration estimators

Another model-assisted approach to estimating  $F_y(t)$  is model calibration [11, 27, 41, 42, 51, 52, 53, 63]. Calibration estimators were first introduced by [13] to estimate population means and totals. For example, the standard design-based estimator of the population total,  $\sum_{i \in s} d_i y_i$ , can be replaced with  $\sum_{i \in s} \omega_i y_i$ , where the  $\omega_i$  are calibrated weights made as close as possible to  $d_i$  while respecting the calibration equation  $\sum_{i \in s} \omega_i x_i = \sum_{i \in N} x_i$ . The calibrated weights are thus perfect fits for the design-based estimator of the population total of  $x$  while maintaining fidelity to the original sampling weights.

Model calibration estimators of the CDF have the form  $\widehat{F}_{MC}(t) = N^{-1} \sum_{i \in s} \omega_i(t) \hat{g}(x_i, t)$ , where  $\omega_i(t)$  are calibrated weights that depend on the point  $t$ , which are also made as close as possible to  $d_i$  but under the constraint  $\sum_{i \in s} \omega_i(t) \hat{g}(x_i, t) = \sum_{i \in U} \hat{g}(x_i, t)$ , where  $\hat{g}(x_i, t)$  is the model-assisted estimate of  $I(y_i \leq t)$ . Thus, if  $\hat{g}(x_i, t)$  closely approximates  $I(y_i \leq t)$  then  $\widehat{F}_{MC}(t)$  might be expected to perform well.

Each of the calibration estimators surveyed in this section are asymptotically design-unbiased and gain efficiency if the assumed model is correctly specified. They differ in their choice of  $\hat{g}(x_i, t)$  and the objective function used to construct an estimator.

### Wu and Sitter estimator

The [63] model-calibrated estimator is obtained by minimizing the chi-square distance function

$$\Phi_s = \sum_{i \in s} \frac{\{\omega_i(t) - d_i\}^2}{d_i q_i}, \quad (2.5)$$

subject to the calibration equations

$$\sum_{i \in s} \omega_i(t) = 1, \quad \sum_{i \in s} \omega_i(t) \hat{g}(x_i, t) = \sum_{i \in U} \hat{g}(x_i, t),$$

where  $q_i$ ,  $i \in s$ , are known positive weights unrelated to  $d_i$  (typically  $q_i = 1$ ,  $i \in s$ ), and  $\hat{g}(x_i, t)$  is one of the previously defined estimators. One benefit of this approach is that resulting CDF estimator can be expressed in closed form, it equals

$$\hat{F}_{\text{MC|WS}}(t) = N^{-1} \sum_{i \in s} d_i I(y_i \leq t) + \left\{ N^{-1} \sum_{i \in U} \hat{g}(x_i, t) - N^{-1} \sum_{i \in s} d_i \hat{g}(x_i, t) \right\} \hat{B},$$

where

$$\hat{B} = \sum_{i \in s} d_i \{ \hat{g}(x_i, t) - \bar{g}(t) \} \{ I(y_i \leq t) - \bar{I}(t) \} / \sum_{i \in s} d_i \{ \hat{g}(x_i, t) - \bar{g}(t) \}^2,$$

$\bar{g}(t) = \sum_{i \in s} d_i \hat{g}(x_i, t) / \sum_{i \in s} d_i$ , and  $\bar{I}(t) = \sum_{i \in s} d_i I(y_i \leq t) / \sum_{i \in s} d_i$ . Because the calibrated weights,  $\omega_i(t)$ , depend on  $t$ , this estimator is not strictly monotone increasing and may take values outside  $[0, 1]$ . We denote  $\hat{Q}_{y, \text{MC|WS}}^{(1)}(\alpha)$  and  $\hat{Q}_{y, \text{MC|WS}}^{(2)}(\alpha)$  to be the estimated quantiles based on inverting a transformed  $\hat{F}_{\text{MC|WS}}(t)$ , using  $\hat{g}(x_i, t) = \hat{G}[\{t - \hat{\mu}(x_i)\} / \hat{\sigma}(x_i)]$  and  $\hat{g}(x_i, t) = I\{\hat{\mu}(x_i) \leq t\}$  respectively. The estimator  $\hat{Q}_{y, \text{MC|WS}}^{(2)}(\alpha)$  has the property that if  $y_i = \mu(x_i)$ ,  $i \in U$ , then  $\hat{Q}_{y, \text{MC|WS}}^{(2)}(\alpha) = Q_y(\alpha)$ .

### Rueda estimator

Instead of calculating a new set of weights,  $\omega_i(t)$ , for each  $t$ , the Rueda estimator [54] calibrates at a fixed set of points,  $\mathbf{t}_0 = \{t_1, t_2, \dots, t_P\}^T$  to produce a single set of weights,  $\omega_i(\mathbf{t}_0)$ ,  $i \in s$ . Let  $\hat{\mu}(x_i)$ ,  $i \in U$ , be a set of fitted values based on an assumed model for  $\mu(x)$ . The weights are obtained by minimizing (3.4) subject to the calibration equations

$$\sum_{i \in s} \omega_i(\mathbf{t}_0) I\{\hat{\mu}(x_i) \leq t_j\} = \sum_{i \in U} I\{\hat{\mu}(x_i) \leq t_j\}, \quad j = 1, \dots, P.$$

The resulting estimator of the CDF is

$$\hat{F}_{y, \text{MC|R}}(t) = \hat{F}_{y, \text{HT}}(t) + \left\{ F_{\hat{\mu}}(\mathbf{t}_0) - \hat{F}_{\hat{\mu}, \text{HT}}(\mathbf{t}_0) \right\} \hat{\mathbf{B}},$$

where

$$F_{\hat{\mu}}(\mathbf{t}_0) = \left[ N^{-1} \sum_{i \in U} I\{\hat{\mu}(x_i) \leq t_1\}, \dots, N^{-1} \sum_{i \in U} I\{\hat{\mu}(x_i) \leq t_P\} \right]$$

and

$$\widehat{F}_{\widehat{\mu},\text{HT}}(\mathbf{t}_0) = \left[ N^{-1} \sum_{i \in s} d_i I\{\widehat{\mu}(x_i) \leq t_1\}, \dots, N^{-1} \sum_{i \in s} d_i I\{\widehat{\mu}(x_i) \leq t_P\} \right]$$

are the population and sample multivariate CDFs of the  $\widehat{\mu}(x_i)$  fitted values; and

$$\widehat{\mathbf{B}} = \mathbf{T}^{-1} \sum_{i \in s} d_i q_i I\{\widehat{\mu}(x_i) \leq \mathbf{t}_0\} I\{y_i \leq \mathbf{t}_0\},$$

where

$$\mathbf{T} = \sum_{i \in s} d_i q_i I\{\widehat{\mu}(x_i) \leq \mathbf{t}_0\} I\{\widehat{\mu}(x_i) \leq \mathbf{t}_0\}^T,$$

$I\{\widehat{\mu}(x_i) \leq \mathbf{t}_0\} = [I\{\widehat{\mu}(x_i) \leq t_1\}, \dots, I\{\widehat{\mu}(x_i) \leq t_P\}]^T$ ,  $I\{y_i \leq \mathbf{t}_0\} = [I\{y_i \leq t_1\}, \dots, I\{y_i \leq t_P\}]^T$ , and  $q_i$ ,  $i \in s$ , are known positive weights unrelated to  $d_i$  (typically  $q_i = 1$ ,  $i \in s$ ). This estimator is asymptotically design-unbiased and guaranteed to be a proper CDF if  $q_i = c$ ,  $i \in s$  for some constant  $c$  and  $t_P$  is chosen large enough such that  $N^{-1} \sum_{i \in s} I\{\widehat{\mu}(x_i) \leq t_P\} = 1$ . Rueda suggests choosing  $t_P = \max_{i \in U} \widehat{\mu}(x_i)$ . The estimator  $\widehat{F}_{y,\text{MC|R}}(t)$  is designed to have low mean squared error at  $\mathbf{t}_0$  but may be inefficient at points away from  $\mathbf{t}_0$ . Thus, care should be taken when choosing the set of calibration points. Let  $\widehat{Q}_{y,\text{MC|R}}(\alpha)$  be the quantile obtained by inverting  $\widehat{F}_{y,\text{MC|R}}(t)$ .

### Post-stratified model-calibrated estimator

The post-stratified model-calibrated estimator [42] is a hybrid of the design-based post stratification estimator and the Rueda estimator. Instead of partitioning  $U$  by values of  $x$ ,  $U$  is partitioned by a set of fitted values  $\widehat{\boldsymbol{\mu}} = \{\widehat{\mu}(x_1), \dots, \widehat{\mu}(x_N)\}$  and model-calibration occurs within each partition. This estimator is motivated by the assumption that  $\widehat{F}_{y,\text{MCR}}(t)$  performs best if  $y_i$ ,  $i \in s$ , is balanced across the values of  $\mu(x_i)$ ,  $i \in U$ . Let  $U_1, U_2, \dots, U_G$ , be the poststrata such that  $i \in U_g$  if and only if  $\widehat{\mu}^{(g-1)} < \widehat{\mu}_i \leq \widehat{\mu}^{(g)}$ , where  $-\infty = \widehat{\mu}^{(0)} < \widehat{\mu}^{(1)} < \widehat{\mu}^{(2)} < \dots < \widehat{\mu}^{(G-1)} < \widehat{\mu}^{(G)} = \infty$ . The CDF estimator is  $\widehat{F}_{y,\text{PSMC}}(t) = N^{-1} \sum_{g \in G} \sum_{i \in s_g} \omega_{ig}(\mathbf{t}_0^g) I(y_i \leq t)$ , where the calibrated weights  $\omega_{ig}(\mathbf{t}_0^g)$  are obtained by minimizing  $\Phi_{s_g} = \sum_{i \in s_g} [\{\omega_{ig}(\mathbf{t}_0^g) - d_{ig}\}/d_{ig}]$  for  $g = 1, \dots, G$ , subject to the calibration equations

$$\sum_{i \in s_g} \omega_{ig} I\{\widehat{\mu}(x_i) \leq t_j^g\} = N_g^{-1} \sum_{i \in U_g} I\{\widehat{\mu}(x_i) \leq t_j^g\}, \quad j = 1, \dots, P, \quad g = 1, \dots, G.$$

The resulting closed-form expression for the CDF estimator is

$$\widehat{F}_{y,\text{PSMC}}(t) = \sum_{g=1}^G \frac{N_g}{N} \left[ \widehat{F}_{y,\text{HT}}^g(t) + \left\{ F_{\widehat{\mu}}^g(\mathbf{t}_0^g) - \widehat{F}_{\widehat{\mu},\text{HT}}^g(\mathbf{t}_0^g) \right\} \widehat{\mathbf{B}}_g \right],$$

where  $N_g$  is the size of  $U_g$ ,  $\mathbf{t}_0^g = \{t_1^g, t_2^g, \dots, t_{P_g}^g\}^T$  are  $P_g$  prespecified points such that  $t_1^g \leq t_2^g \leq \dots \leq t_{P_g}^g$ , and  $F_{\hat{\mu}}^g(\mathbf{t}_0^g)$ ,  $\hat{F}_{\hat{\mu}, \text{HT}}^g(\mathbf{t}_0^g)$ ,  $\hat{\mathbf{B}}_g$ ,  $\mathbf{T}_g$ , and  $I\{\hat{\mu}(x_i) \leq \mathbf{t}_0^g\}$ , are defined analogously to the corresponding quantities in the Rueda estimator but computed within each stratum (see [42] for exact derivations). Similar to the Rueda estimator,  $\hat{F}_{\text{PSMC}}(t)$  is only guaranteed to be a proper CDF if  $t_P^g$  is chosen large enough such that  $N^{-1} \sum_{i \in s_g} I\{\hat{\mu}(x_i) \leq t_P^g\} = 1$  for  $g = 1, \dots, G$ . [42] suggest choosing  $t_P^g = \max_{i \in U_g} \hat{\mu}(x_i)$ ,  $g = 1, \dots, G$ . Let  $\hat{Q}_{y, \text{PSMC}}(\alpha)$  be the quantile estimator obtained by inverting  $\hat{F}_{y, \text{PSMC}}(t)$ .

### Wu and Sitter empirical likelihood calibration estimators

The Wu and Sitter empirical likelihood calibration estimators have the form

$$\hat{F}_{y, \text{EL}|\text{WS}}(t) = \sum_{i \in s} \hat{\omega}_i(t) I(y_i \leq t),$$

where  $\hat{\omega}_i(t)$  are obtained by maximizing the empirical log-likelihood function

$$\hat{l}(\boldsymbol{\omega}(t)) = \sum_{i \in s} d_i \log \omega_i(t)$$

subject to

$$\sum_{i \in s} \omega_i(t) = 1, \quad \sum_{i \in s} \omega_i(t) \hat{g}(x_i, t) = N^{-1} \sum_{i \in U} \hat{g}(x_i, t),$$

for some fitted values  $\hat{g}(x_i, t)$ . Unlike the previous model-calibrated estimators, the weights do not have a closed form solution but can be represented as  $\hat{\omega}_i(t) = d_i / \{1 + \lambda \hat{g}(x_i, t)\}$  where  $\lambda$  is the solution to  $\sum_{i \in s} d_i \hat{g}(x_i, t) / \{1 + \lambda \hat{g}(x_i, t)\} = 0$ . The resulting CDF will only take values in  $[0, 1]$ , but because the estimated weights depend on  $t$ , the CDF is not guaranteed to be monotone. Define  $\hat{Q}_{y, \text{EL}|\text{WS}}^{(1)}(\alpha)$  and  $\hat{Q}_{y, \text{EL}|\text{WS}}^{(2)}(\alpha)$  to be the estimated quantiles based on inverting the transformed  $\hat{F}_{y, \text{EL}|\text{WS}}(t)$ , using  $\hat{g}(x_i, t) = \hat{G}[\{t - \hat{\mu}(x_i)\} / \hat{\sigma}(x_i)]$  and  $\hat{g}(x_i, t) = I\{\hat{\mu}(x_i) \leq t\}$  respectively. One reason for using  $\hat{Q}_{y, \text{EL}|\text{WS}}^{(2)}(\alpha)$  is that if  $y_i = \mu(x_i)$ ,  $i \in U$ ,  $\hat{Q}_{y, \text{EL}|\text{WS}}^{(2)}(\alpha) = Q_y(\alpha)$ .

### Chen and Wu empirical likelihood calibration estimators

In order to produce an empirical likelihood calibration estimator that is a proper CDF, calibration can occur at a fixed point  $t_0$ , thus producing a single set of estimated weights  $\hat{\omega}_i(t_0)$ ,  $i \in s$ . The [11] empirical likelihood estimators have the form  $\hat{F}_{y, \text{EL}|\text{CW}}(t) = \sum_{i \in s} \hat{\omega}_i(t_0) I(y_i \leq t)$ , where  $\hat{\omega}_i(t_0)$ ,  $i \in s$ , are obtained in the same manner as the Wu and Sitter estimators, with the only difference being that the calibrated weights are estimated once at  $t_0$ . The resulting CDF is a proper CDF and is designed to have low mean squared error in a neighborhood of  $t_0$  but may be inefficient away from  $t_0$ . The three choices of  $\hat{g}(x_i, t)$  proposed by

Chen and Wu (2002) are: (i)  $\hat{g}(x_i, t) = \hat{G}[\{t - \hat{\mu}(x_i)\}/\hat{\sigma}(x_i)]$ , under an assumed location-scale model; (ii)  $\hat{g}(x_i, t) = I\{\hat{\mu}(x_i) \leq t\}$  under a model for  $\mathbb{E}(y_i|x_i) = \mu(x_i)$ ; and (iii)  $\hat{g}(x_i, t) = \exp\{\hat{m}(x_i)\}/[1 + \exp\{\hat{m}(x_i)\}]$ , under the logistic regression model

$$\log([\mathbb{E}\{I(y_i \leq t)|x_i\}]/[1 - \mathbb{E}\{I(y_i \leq t)|x_i\}]) = m(x_i),$$

$i \in U$ , for some smooth function  $m(x)$ . For each choice of  $\hat{g}(x_i, t)$ , the estimator of the CDF is well defined and we denote the corresponding quantile estimators as  $\hat{Q}_{y,EL|CW}^{(1)}(\alpha)$ ,  $\hat{Q}_{y,EL|CW}^{(2)}(\alpha)$ , and  $\hat{Q}_{y,EL|CW}^{(3)}(\alpha)$ , respectively. Both  $\hat{Q}_{y,EL|CW}^{(1)}(\alpha)$  and  $\hat{Q}_{y,EL|CW}^{(2)}(\alpha)$  can be extended to calibrate on  $\mathbf{t}_0 = \{t_1, t_2, \dots, t_P\}$  [52].

## 2.3 Simulation study

We conducted a simulation study to compare the finite-sample performance of the surveyed estimators. We considered finite populations of size  $N = 2000$  generated from the following superpopulation models:

$$\begin{aligned} \text{M1} & : y_i = x_i + \sigma\epsilon_i; \\ \text{M2} & : y_i = x_i + \sigma x_i \epsilon_i; \\ \text{M3} & : y_i = \log(x_i) + \sigma\epsilon_i; \\ \text{M4} & : y_i = \{0.35 + 2(x_i - 0.5)\} I(x_i \leq 0.65) + \sigma\epsilon_i. \end{aligned}$$

Plots of the finite populations are displayed in Figure 2.1. The  $x_i$  values were assumed to be known and are generated i.i.d. from a Uniform(0,1). The  $\epsilon_i$  values were generated i.i.d. from a standard normal distribution with mean 0 and standard deviations of  $\sigma = 0.1$  or  $\sigma = 0.3$ , resulting in  $\text{Corr}(x, y) = 0.95$  and  $0.8$  respectively under M1.

In M1,  $y$  and  $x$  are linearly related, and thus estimators involving a linear model assumption for  $\mathbb{E}(y_i|x_i)$  should perform well. M2 is similar to the first but with heteroscedastic errors allowing us to study the impact a misspecified  $\sigma(x)$ . M3 has a smooth but non-linear mean whereas M4 has a discontinuous mean function.

We considered three sample sizes  $n = 25$ ,  $n = 50$ , and  $n = 100$ , and two sampling schemes, simple random sampling without replacement (SRSWOR) and Poisson sampling. Under Poisson sampling, the inclusion probabilities were proportional to  $x_i$  and sample sizes were *expected* sample sizes because the exact number of samples varies across Monte-Carlo replications. Note that under (SRSWOR), the inclusion probabilities are constant and are  $\pi_i = nN^{-1}$  and  $\pi_{ij} = n(n-1)[N(N-1)]^{-1}$ , for  $i, j \in s$ . Under Poisson sampling,  $\pi_i = nx_i (\sum_{i \in U} x_i)^{-1}$  and  $\pi_{ij} = \pi_i \pi_j$  for  $i, j \in s$ .

Let  $\tilde{F}_y(t)$  be any of the discussed CDF estimators. In order to invert  $\tilde{F}_y(t)$ , we first computed  $\tilde{F}_y(t_1), \tilde{F}_y(t_2), \dots, \tilde{F}_y(t_{1000})$  where  $t_1 < t_2 < \dots < t_{1000}$  are equally-spaced points spanning the range of values of  $y$ . If any of the  $\tilde{F}_y(t_1), \tilde{F}_y(t_2), \dots, \tilde{F}_y(t_{1000})$  values were outside of  $[0,1]$ , we set them to the closest boundary. Next, if  $\tilde{F}_y(t_1), \tilde{F}_y(t_2), \dots, \tilde{F}_y(t_{1000})$  was not a nondecreasing set, we used the Pool Adjacent Violators Algorithm [49] to obtain a strictly nondecreasing set. We then defined the corresponding quantile estimator,  $\tilde{Q}_y(\alpha)$ , to be the smallest  $t_i$  for which  $\tilde{F}_y(t_i) \geq \alpha$ .

For each combination of sampling design,  $n$ ,  $\sigma$ , and generative model, we generated  $M = 1000$  replicate samples. For each sample  $i = 1, \dots, M$ , we computed  $\tilde{Q}_y(\alpha)^{(i)}$  and the squared error  $\left\{ \tilde{Q}_y(\alpha)^{(i)} - Q_y(\alpha) \right\}^2$  for each  $\alpha \in \boldsymbol{\alpha} = \{0.1, 0.2, 0.3, 0.4, 0.5, 0.6, 0.7, 0.8, 0.9\}$ . We then computed the average squared error (ASE) defined as

$$\text{ASE} \left( \tilde{Q}_y \right)^{(i)} = (1/9) \sum_{\alpha \in \boldsymbol{\alpha}} \left\{ \tilde{Q}_y(\alpha)^{(i)} - Q_y(\alpha) \right\}^2$$

. Lastly, we computed the mean average squared error (MASE), defined as

$$\text{MASE}(\tilde{Q}_y) = M^{-1} \sum_{i=1}^M \text{ASE} \left( \tilde{Q}_y \right)^{(i)}.$$

In preliminary simulation studies, we investigated the different choices of nonparametric weighting schemes. There was not a large difference in performance among the estimators so we selected the local-linear regression estimator for  $\mu(x)$  and the normal kernel density estimator everywhere else. For the density estimator,  $\hat{Q}_{y,D|DEN}$ , we followed [33] and used the logistic density function and CDF in the conditional kernel density estimator. For all kernel-based estimators, we used the plug-in bandwidth  $b = 1.06\hat{\sigma}_x n^{-1/5}$ , where  $\hat{\sigma}_x$  is the sample standard deviation of the  $x$  values, proposed by [59].

Of the model-calibrated estimators, we did not include the Wu and Sitter estimators as they are not proper CDFs while the similar Rueda and Chen and Wu estimators are. The Rueda estimator was calibrated on  $\mathbf{t} = \{Q_{\hat{\mu}}(\alpha), Q_{\hat{\mu}}(1)\}^T$ , the  $\alpha$  quantile of a set of fitted values along with the maximum fitted value. The first point is because  $\hat{F}_{y,MC|R}(t)$  is most efficient at the point of calibration, while the second point ensures that  $\hat{F}_{y,MC|R}(t)$  is a proper CDF [54]. The post-stratified model-calibrated estimator was calibrated at the median and maximum value of  $\hat{\mu}$  within each poststrata [42]. We encountered problems solving for the calibration weights when too many calibration points were used or when the points were too close together. For the empirical likelihood estimators, calibration always occurred at  $t_0 = Q_{\hat{\mu}}(\alpha)$  because the CDF estimators are most efficient at the point of calibration. In preliminary simulation studies,  $\hat{Q}_{y,EL|CW}^{(3)}(\alpha)$  often could not be computed because the generalized linear model relating

$I(y_i \leq t_0)$  and  $x_i$  for  $i \in s$  would not converge. This occurred in instances where  $I(y_i \leq t_0) = 1$  for all  $x_i$  larger than some value and 0 otherwise. We therefore omitted it from the analysis.

The post-stratification and post-stratified model-calibrated estimators were obtained using  $G = 4$  equally spaced poststrata. If there was any empty poststrata, then  $G$  was reduced by 1 until there were no empty poststrata.

For model-based and model-assisted estimators, we add either a ‘p’ (‘parametric’) or ‘np’ (‘nonparametric’) subscript to denote that  $\mathbb{E}(y|x)$  was estimated by  $\hat{\mu}(x) = \hat{\beta} + \hat{\beta}_1 x$  or  $\hat{\mu}(x) = \sum_{i \in s} \omega_i(x) y_i$  respectively. We assumed constant variance for all model-based and model-assisted estimators with the only exception being  $\hat{Q}_{y,M|LGP}(\alpha)$ , in which the variance was estimated using nonparametric methods.

We did not include the hybrid  $\hat{Q}_{y,H}(\alpha)$  estimator in the simulation study because [62] only provided a derivation of the CDF estimator under simple random sampling and a simple linear parametric superpopulation model.

For brevity, when discussing the quantile estimators, we remove the  $y$  subscript and ‘ $(\alpha)$ ’ from their titles, for instance,  $\hat{Q}_{M|CD}$  denotes the Chambers and Dunstan estimator  $\hat{Q}_{y,M|CD}(\alpha)$ .

### 2.3.1 Results

#### Design-based estimators

The MASE values and corresponding standard errors of the design-based estimators are displayed in Table 2.2 for both SRS and Poisson sampling for  $n = 100$  and  $\sigma = 0.1$ . Boxplots showing the distribution of ASE values are displayed in Figures 2.2 and 2.3. The ratio estimator  $\hat{Q}_{D|R}$ , position estimator  $\hat{Q}_{D|POS}$ , and both direct quantile-based estimators  $\hat{Q}_{D|R}^*$  and  $\hat{Q}_{D|D}^*$  are not included in these results because they performed significantly worse than the other estimators.

In general,  $\hat{Q}_{D|DEN}$  performed best among design-based estimators. Under M1, it had the lowest (or tied for the lowest) ASE values. Under M2,  $\hat{Q}_{D|DEN}$  performed the best if  $\sigma = 0.3$  (higher variance) but  $\hat{Q}_{D|D}$  performed best if  $\sigma = 0.1$ . Under M3 and M4,  $\hat{Q}_{D|DEN}^*$  always had the lowest ASE values with the exception of  $n = 100, \sigma = 0.1$ , and SRS scenario, where  $\hat{Q}_{D|PS}$  performed best.

#### Model-based estimators

The performance of the model-based estimators was more varied across the simulation scenarios. Table 2.3 displays the MASE values for  $n = 100$  and  $\sigma = 0.1$ . The corresponding ASE values are displayed in Figures 2.4 and 2.5. We did not include  $\hat{Q}_{M|CDW_{np}}$  as it performed virtually identically to  $\hat{Q}_{M|CDW_p}$ . We also did not include  $\hat{Q}_{M|K}$  as it always had ASE values that were

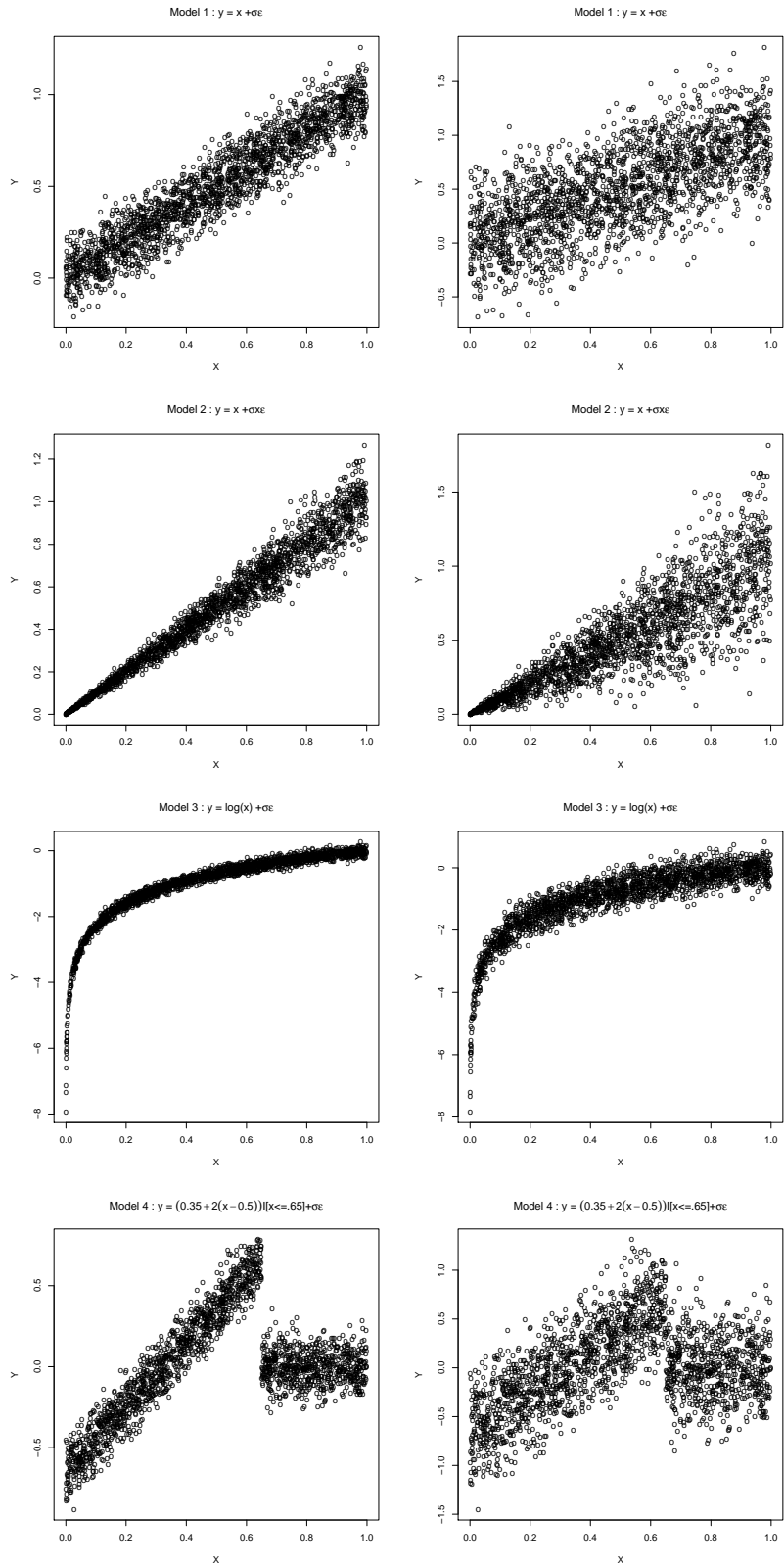


Figure 2.1: Finite populations of size  $N = 2000$  generated from models 1-4 with  $\sigma = 0.1$  (left) and  $\sigma = 0.3$  (right).

Table 2.2: MASE values ( $\times 10^2$ ) and corresponding standard deviations for the design-based estimators with  $n = 100$  and  $\sigma = 0.1$ . The smallest MASE values for each generative model are **bolded**.

Estimator	SRS				Poisson			
	M1	M2	M3	M4	M1	M2	M3	M4
$\hat{Q}_{HTN}$	.20(.18)	.18(.17)	2.19(2.54)	.23(.20)	.68(1.26)	.60(.96)	22.7(77.24)	1.1(2.07)
$\hat{Q}_{DID}$	<b>.07(.03)</b>	<b>.03(.02)</b>	2.19(2.54)	.34(.38)	<b>.15(.24)</b>	<b>.03(.02)</b>	8.42(9.80)	1.4(2.47)
$\hat{Q}_{D PS}$	.07(.04)	.05(.03)	1.02(1.45)	<b>.14(.11)</b>	.17(.20)	.11(.09)	<b>7.6(20.61)</b>	<b>.40(.47)</b>
$\hat{Q}_{D DEN}$	.14(.06)	.11(.05)	<b>.94(.90)</b>	.40(.21)	.20(.22)	.15(.11)	7.9(22.84)	.60(.49)

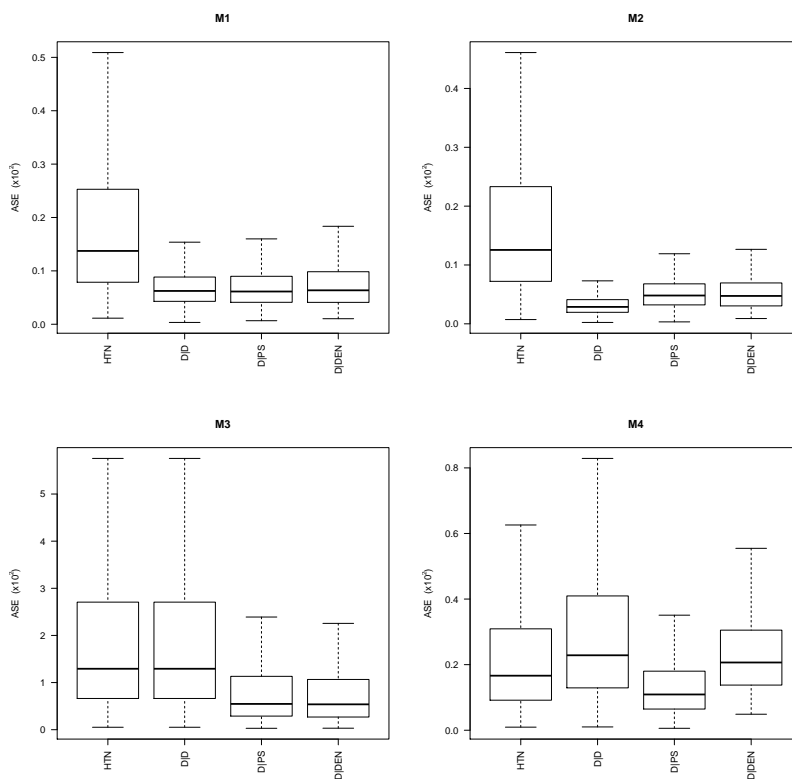


Figure 2.2: Boxplots of ASE values ( $\times 10^2$ ) for the design-based estimators from models 1-4 with  $n = 100$  and  $\sigma = 0.1$  under SRS.

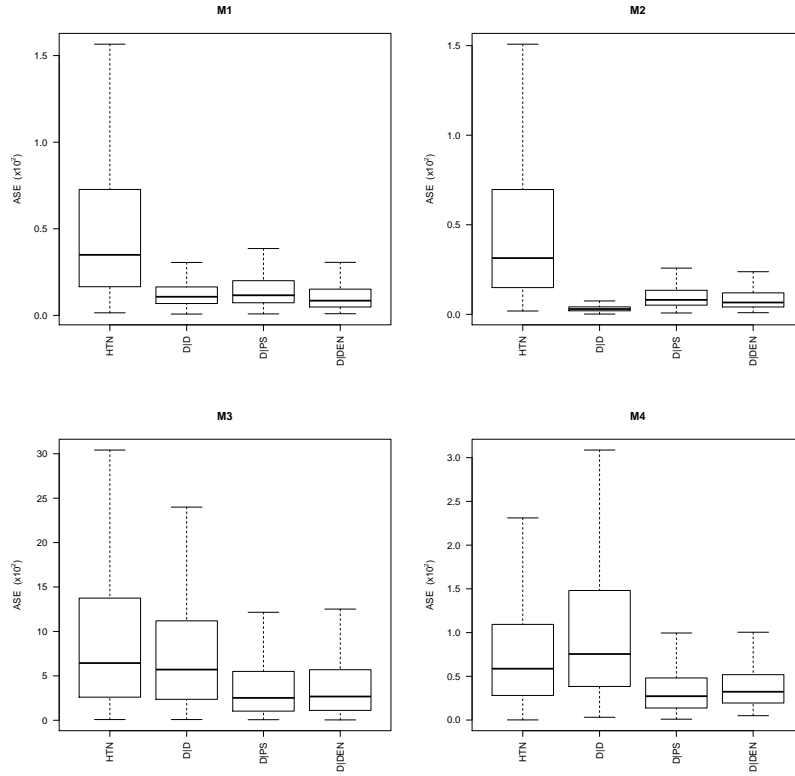


Figure 2.3: Boxplots of ASE values ( $\times 10^2$ ) for the design-based estimators from models 1-4 with  $n = 100$  and  $\sigma = 0.1$  under Poisson sampling.

Table 2.3: MASE values ( $\times 10^2$ ) and corresponding standard deviations for the model-based estimators with  $n = 100$  and  $\sigma = 0.1$ . The smallest MASE values for each generative model are **bolded**.

Estimator	SRS				Poisson			
	M1	M2	M3	M4	M1	M2	M3	M4
$\hat{Q}_{M CD_p}$	<b>.02(.02)</b>	<b>.01(.01)</b>	4.99(1.05)	.29(.14)	<b>.03(.03)</b>	<b>.01(.01)</b>	5.68(1.36)	1.71(.72)
$\hat{Q}_{M LGP_p}$	<b>.02(.02)</b>	<b>.01(.01)</b>	3.09(1.13)	.56(.36)	.04(.03)	<b>.01(.01)</b>	8.71(1.38)	1.06(.80)
$\hat{Q}_{M CD_{np}}$	<b>.02(.02)</b>	<b>.01(.01)</b>	1.53(1.09)	.27(.06)	.07(.08)	<b>.01(.01)</b>	<b>.63(.85)</b>	<b>.27(.10)</b>
$\hat{Q}_{M CDW_p}$	.05(.03)	.03(.01)	<b>.41(.33)</b>	<b>.13(.10)</b>	.08(.06)	.04(.02)	2.81(1.60)	.57(.37)

greater than or equal to  $\hat{Q}_{M|CDW_p}$ , suggesting that including the bias-correction term improved  $\hat{Q}_{M|CDW_p}$ .

Under M1,  $\hat{Q}_{M|CD_p}$  performed best. This is expected as the model is correctly specified. The  $\hat{Q}_{M|CD_{np}}$  estimator on average performed similarly to  $\hat{Q}_{M|CD_p}$  under M2 but always had a higher variance. Under M2,  $\hat{Q}_{M|LGP_p}$  did not always have the lowest ASE values despite involving a correctly specified model. This is likely due to the added variance involved in estimating  $\sigma(x)$ . Overall, there was little evidence supporting  $\hat{Q}_{M|LGP_p}$  because under M1, M2, and M3,  $\hat{Q}_{M|CD_p}$  performed better and under M3  $\hat{Q}_{M|CD_{np}}$  performed better. This suggests that when deciding among model-based estimators, a correctly specified model for  $\mu(x)$  is more critical than a correctly specified model for  $\sigma(x)$ .

Under M3,  $\hat{Q}_{M|CDW_p}$  performed better than  $\hat{Q}_{M|CD_p}$  under SRS in all cases except for  $n = 25$ , while  $\hat{Q}_{M|CD_{np}}$  always performed the best under Poisson sampling. This is likely because under Poisson sampling, it is more likely to sample larger  $x$  values where the curve is more linear, and thus  $\mu(x)$  can be better approximated in that region. Under M4,  $\hat{Q}_{M|CD_{np}}$  was always the best performing estimator under Poisson sampling and was the best performing under SRS for  $\sigma = 0.3$ . If  $\sigma = 0.1$  under SRS,  $\hat{Q}_{M|CDW_p}$  had the lowest ASE values. Both  $\hat{Q}_{M|CD_p}$  and  $\hat{Q}_{M|LGP_p}$  had significantly higher ASE values under M3 and M4, revealing their sensitivity to model misspecification.

### Generalized difference estimators

Results of the generalized difference estimators for  $n = 100$  and  $\sigma = 0.1$  are displayed in Table 2.4 and Figures 2.6 and 2.7. There was lower variance in the ASE values of generalized difference estimators compared with model-based estimators suggesting that generalized difference estimators are more robust to model misspecification. The estimator  $\hat{Q}_{GD|RKM_{np}}$  was generally the best performing of the group in terms of ASE. Under M1,  $\hat{Q}_{GD|RKM_p}$  was always the best performing estimator but under M3 and M4 it performed much worse than  $\hat{Q}_{GD|RKM_{np}}$ .

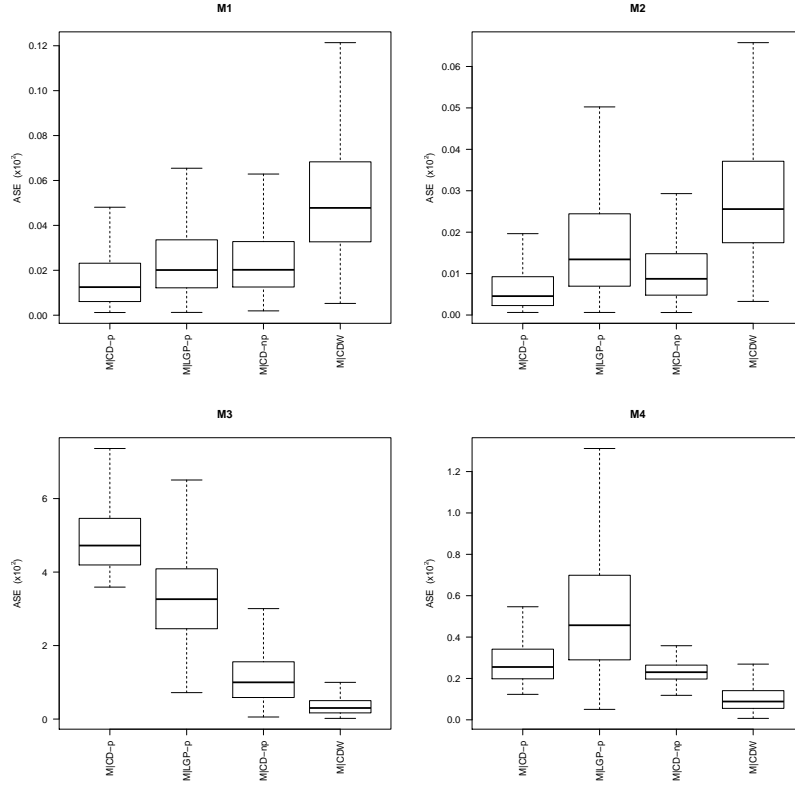


Figure 2.4: Boxplots of ASE values ( $\times 10^2$ ) for the model-based estimators from models 1-4 with  $n = 100$  and  $\sigma = 0.1$  under SRS.

Table 2.4: MASE values ( $\times 10^2$ ) and corresponding standard deviations for the generalized difference estimators with  $n = 100$  and  $\sigma = 0.1$ . The smallest MASE values for each generative model are **bolded**.

Estimator	SRS				Poisson			
	M1	M2	M3	M4	M1	M2	M3	M4
$\hat{Q}_{GD RKM_p}$	<b>.05(.03)</b>	<b>.03(.01)</b>	.64(.76)	.19(.17)	<b>.12(.13)</b>	.04(.02)	7.42(21.62)	.64(.78)
$\hat{Q}_{GD RKM_{np}}$	.05(.03)	<b>.03(.01)</b>	<b>.29(.29)</b>	<b>.09(.06)</b>	.15(.16)	.04(.02)	<b>.78(1.11)</b>	<b>.22(.23)</b>
$\hat{Q}_{GD WS_{np}}$	.07(.04)	.03(.02)	.77(.89)	.26(.22)	.14(.14)	<b>.03(.01)</b>	7.10(17.31)	.85(.81)
$\hat{Q}_{GD K}$	.06(.03)	.04(.02)	.47(.44)	.10(.07)	.14(.14)	.07(.06)	4.63(10.45)	.28(.28)

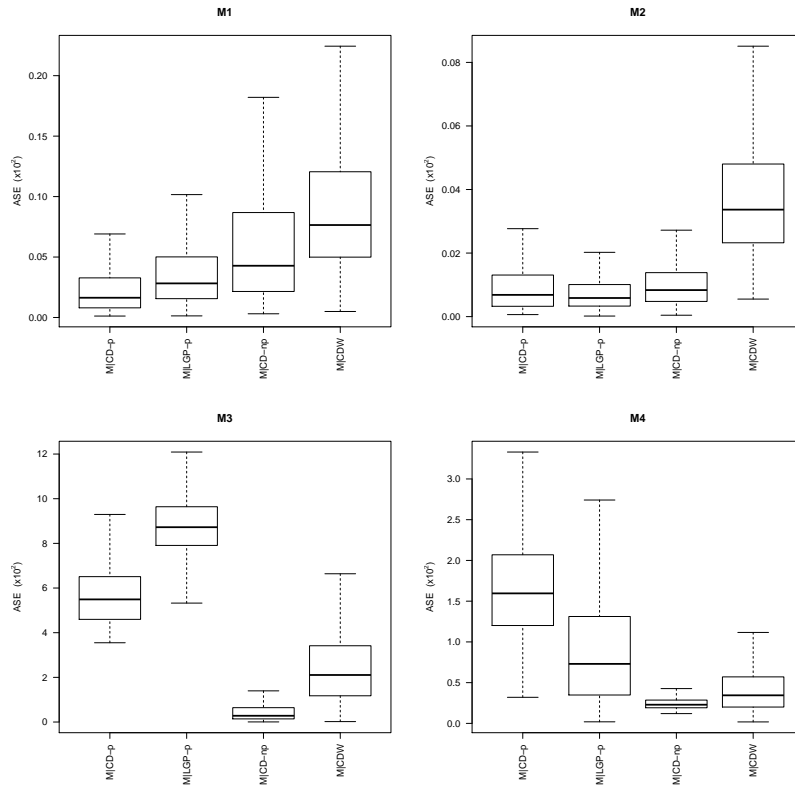


Figure 2.5: Boxplots of ASE values ( $\times 10^2$ ) for the model-based estimators from models 1-4 with  $n = 100$  and  $\sigma = 0.1$  under Poisson sampling.

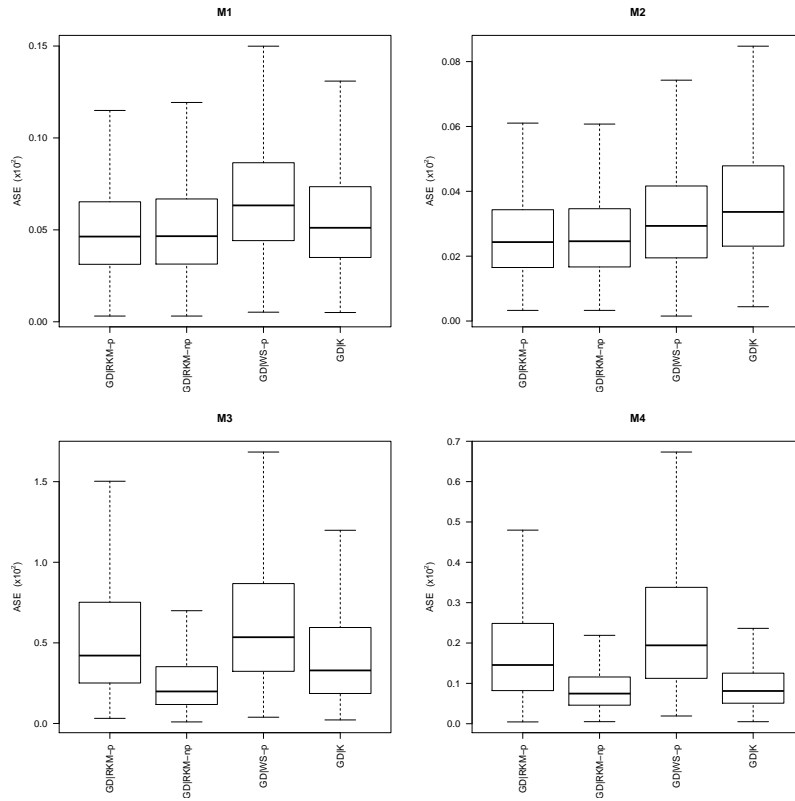


Figure 2.6: Boxplots of ASE values ( $\times 10^2$ ) for the generalized difference estimators from models 1-4 with  $n = 100$  and  $\sigma = 0.1$  under SRS.

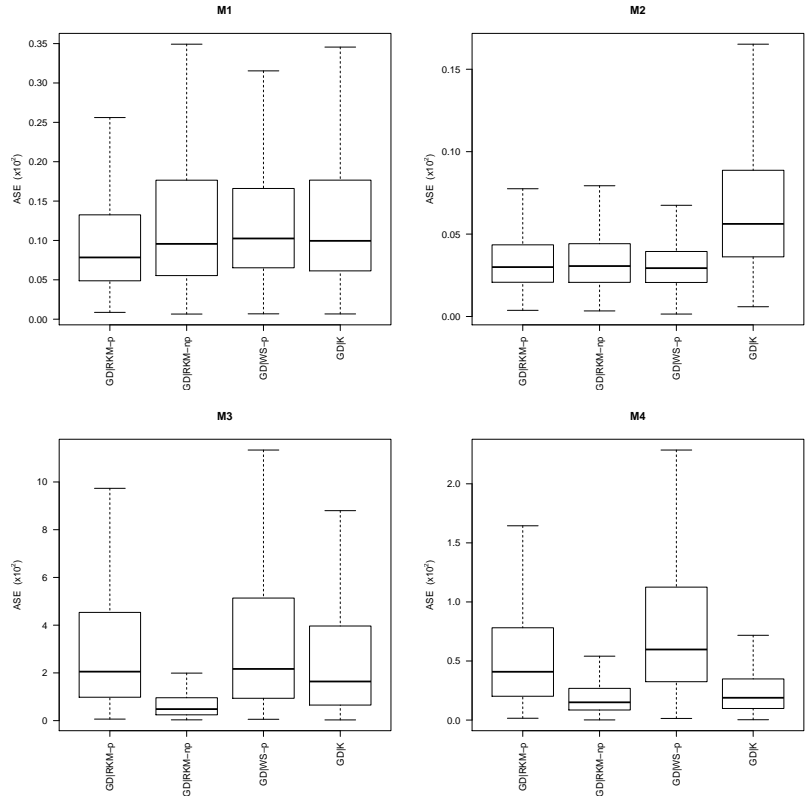


Figure 2.7: Boxplots of ASE values ( $\times 10^2$ ) for the generalized difference estimators from models 1-4 with  $n = 100$  and  $\sigma = 0.1$  under Poisson.

Table 2.5: MASE values ( $\times 10^2$ ) and corresponding standard deviations for the calibration estimators with  $n = 100$  and  $\sigma = 0.1$ . The smallest MASE values for each generative model are **bolded**.

Estimator	SRS				Poisson			
	M1	M2	M3	M4	M1	M2	M3	M4
$\widehat{Q}_{MC R_p}$	.07(.04)	.04(.02)	.44(.69)	.15(.12)	.21(.20)	.13(.14)	5.03(7.36)	.55(.48)
$\widehat{Q}_{PSMC_p}$	.06(.04)	.04(.02)	.51(.64)	<b>.08(.05)</b>	.16(.14)	.07(.06)	4.17(10.43)	.28(.29)
$\widehat{Q}_{EL CW_p}^{(1)}$	<b>.05(.03)</b>	<b>.03(.02)</b>	.47(.52)	.19(.16)	.15(.15)	<b>.06(.09)</b>	5.49(18.72)	.60(.66)
$\widehat{Q}_{EL CW_{np}}^{(1)}$	<b>.05(.03)</b>	<b>.03(.02)</b>	<b>.20(.17)</b>	.09(.06)	<b>.15(.14)</b>	<b>.06(.09)</b>	<b>2.49(3.54)</b>	<b>.26(.29)</b>

### Calibration estimators

Results for the calibration estimators are displayed in Table 2.5 and Figures 2.8 and 2.9. The  $\widehat{Q}_{MC|R}$  and  $\widehat{Q}_{PSMC}$  estimators had approximately the same ASE values whether  $\mu(x)$  was estimated nonparametrically or parametrically, thus only the parametric versions were included in the results. Of the empirical likelihood estimators,  $\widehat{Q}_{EL|CW}^{(2)}(\alpha)$  never outperformed  $\widehat{Q}_{EL|CW}^{(1)}$  and was thus also not included in the results.

In general, the  $\widehat{Q}_{EL|CW_{np}}^{(1)}$  was the best performing of these estimators. It never performed significantly worse than  $\widehat{Q}_{EL|CW_p}^{(1)}$  but performed better under M3 and M4. The post-stratified model-calibrated estimator performed particularly well under Model 4.

### Comparing the estimators with generally the lowest ASE values from each estimator category

Figures 2.10 and 2.11 display the estimators with the lowest (in general) ASE values from each of the design-based, model-based, and model-assisted categories under both sampling schemes. The results shown are based on  $\sigma = 0.1$ , and  $n = 100$ . The other simulation scenarios produced similar results.

Under Models 1 and 2, all of the estimators significantly outperformed  $\widehat{Q}_{HTN}$ , with  $\widehat{Q}_{M|CD_p}$  having the lowest ASE values followed closely by  $\widehat{Q}_{M|CD_{np}}$ . Under model 3,  $\widehat{Q}_{M|CD_p}$  did particularly poorly, having significantly higher ASE values. The remaining estimators always performed better than  $\widehat{Q}_{y,HTN}$  with  $\widehat{Q}_{M|CWD_p}$ ,  $\widehat{Q}_{GD|RKM_{np}}$ , and  $\widehat{Q}_{MC|CD_{np}}$  having the smallest ASE values. Under Model 4,  $\widehat{Q}_{M|CWD_p}$ ,  $\widehat{Q}_{GD|RKM_{np}}$ , and  $\widehat{Q}_{EL|CW_{np}}^{(1)}$  performed best.

These results show that if the relationship between  $x$  and  $y$  is linear, then  $\widehat{Q}_{M|CD_p}$  or  $\widehat{Q}_{M|CD_{np}}$  should be used, with  $\widehat{Q}_{M|CD_{np}}$  being the more robust of the two. If the relationship between  $x$  and  $y$  is strongly nonlinear then  $\widehat{Q}_{GD|RKM_{np}}$  or  $\widehat{Q}_{EL|CW_{np}}^{(1)}$  should be used. While  $\widehat{Q}_{D|DEN}$  was generally the best performing of the design-based estimators, it was always outperformed by one of the model-assisted or model-based estimators in every scenario, even when

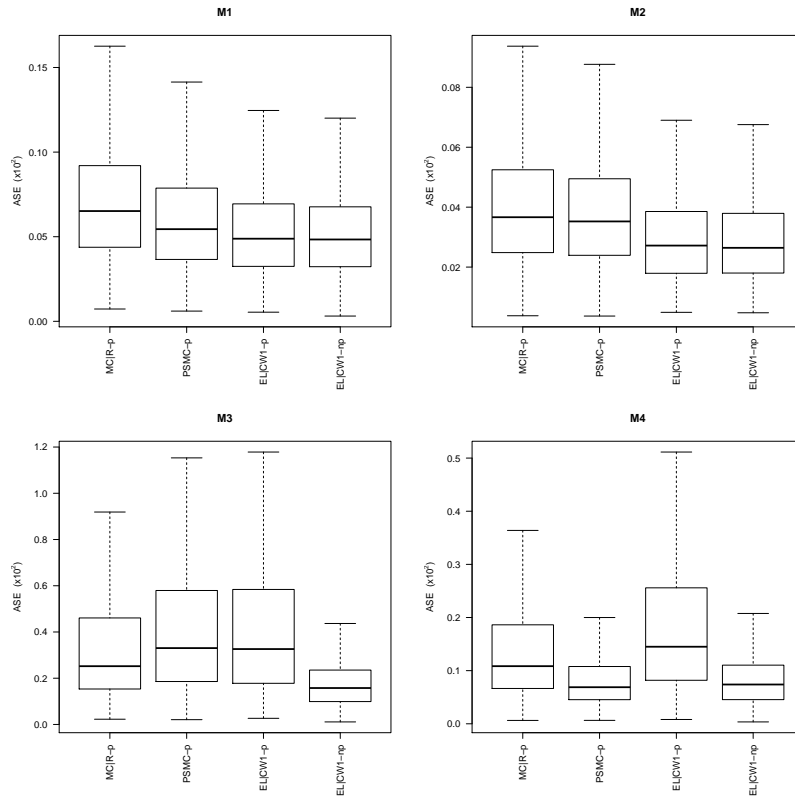


Figure 2.8: Boxplots of ASE values ( $\times 10^2$ ) for the calibration estimators from models 1-4 with  $n = 100$  and  $\sigma = 0.1$  under SRS

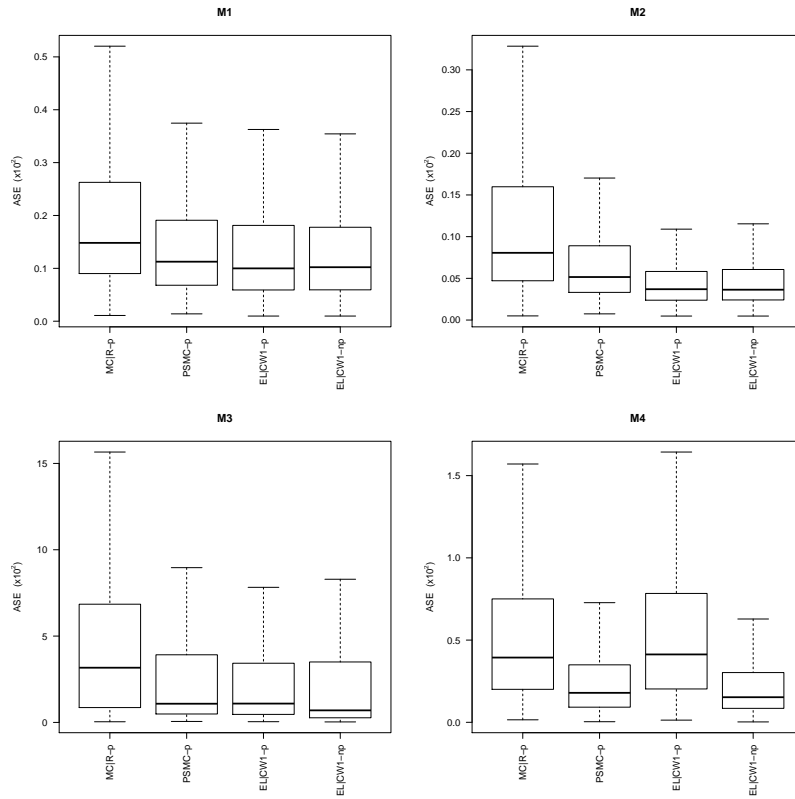


Figure 2.9: Boxplots of ASE values ( $\times 10^2$ ) for the calibration estimators from models 1-4 with  $n = 100$  and  $\sigma = 0.1$  under Poisson sampling.

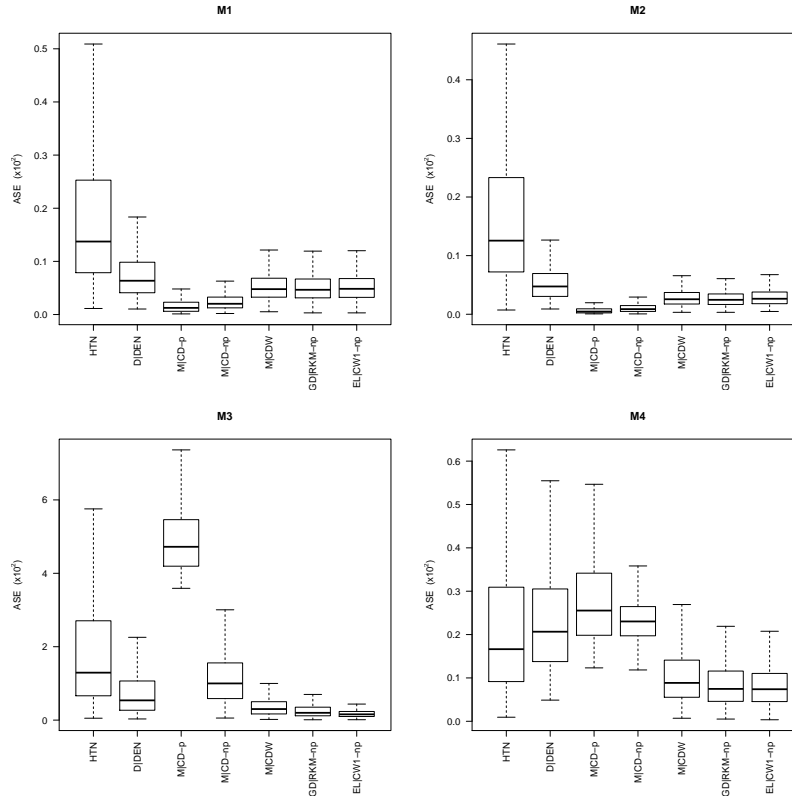


Figure 2.10: Boxplots of ASE values ( $\times 10^2$ ) from models 1-4 with  $n = 100$  and  $\sigma = 0.1$  under SRS

the models were misspecified.

Thus far, we have focused on the performance of the estimators averaged over values of  $\alpha$ . In Figure 2.12 we display the MSE values of four estimators for each  $\alpha$  when  $n = 100, \sigma = 0.1$ , under SRS. The estimators that performed the best on average were typically the best across all quantiles. In general, the quantiles near the center of the distribution were estimated more efficiently than those on the tails.

## 2.4 Conclusion

Quantile estimation using auxiliary information has been studied extensively in survey sampling, however, exposure to these methods is limited in most current statistical training programs. This is unfortunate as these methods are well-suited to the analysis of large observational heterogeneous data sources frequently arising in modern ‘big-data’ applications. The purpose of this survey was to present these methods, which had heretofore been scattered across the statis-

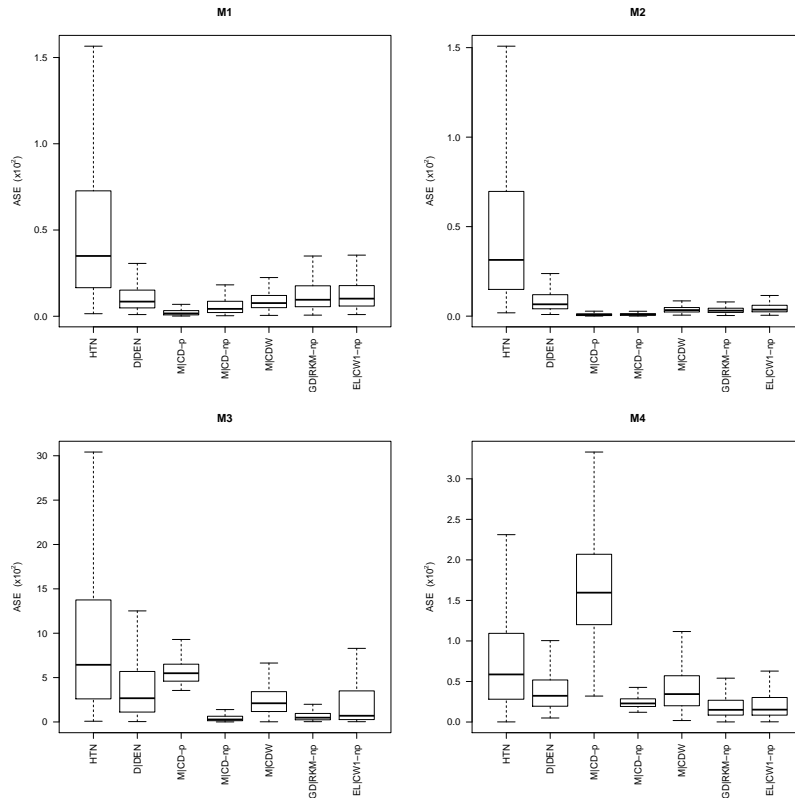


Figure 2.11: Boxplots of ASE values ( $\times 10^2$ ) from models 1-4 with  $n = 100$  and  $\sigma = 0.1$  under Poisson sampling

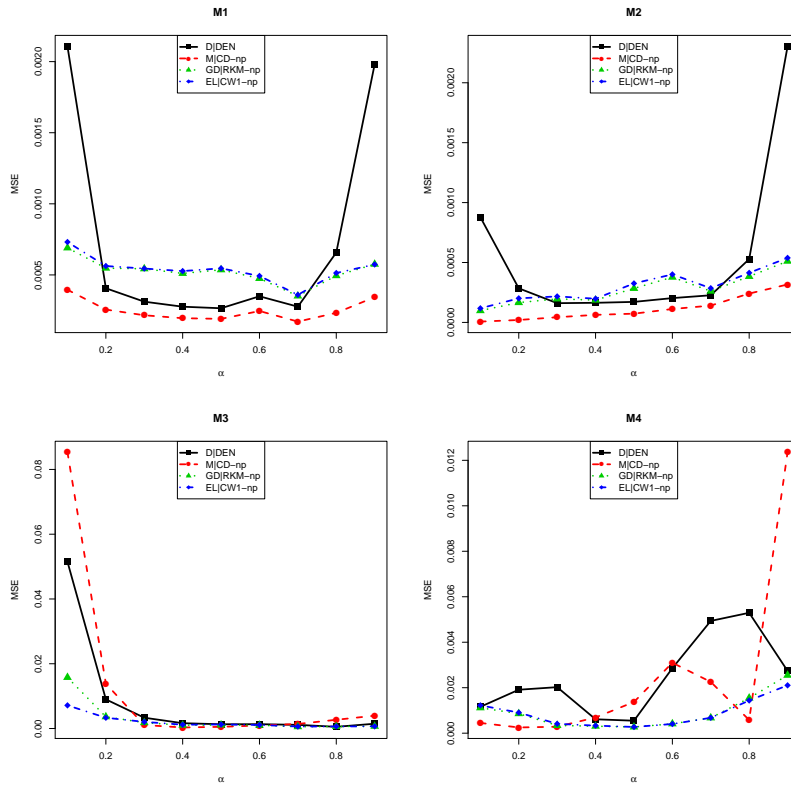


Figure 2.12: Plots of MSE values across increasing values of  $\alpha$  for  $n = 100$  and  $\sigma = 0.1$  under SRS.

tics and survey sampling literature, in a single accessible reference. In addition, we conducted simulation experiments with the most complete set of quantile estimators using auxiliary information to date. We anticipate that survey sampling methods like the one discussed here will play a prominent role in statistical thinking for data science and big data analytics.

## Chapter 3

# R Package for Auxiliary Quantile Estimation

### 3.1 Introduction

Estimating a population quantile is a standard task in statistical practice. Often, the mean and standard deviation are not sufficient to summarize a distribution effectively. Common quantiles of interest are the 25th, 50th (median), and 75th used to obtain the interquantile range or the more extreme quantiles such as the 1st, 5th, 95th, or 99th used to construct confidence intervals. Compared to the mean and standard deviation, quantiles (particularly those near the tails of the distribution) require a larger sample size to obtain precise estimates. If the sample size is limited, quantile estimators may perform poorly.

Consider the scenario where there exists a correlated auxiliary variable that is observed on all units in the population. There exists a broad class of quantile estimators which incorporate this auxiliary information which can often improve upon standard estimators, particularly when the correlation between the variables is high. We denote estimators of this form as ‘auxiliary estimators’. Auxiliary estimation is a form of semi-supervised learning [10, 64, 1] in which unlabeled data is used along with a small set of labeled data to understand an underlying relationship in the data. Auxiliary estimators of the population quantile, under the heading of quantile estimation with auxiliary information, have been studied extensively in the literature and have been compared in detail in [20].

We introduce the **R** [61] package **auxQuantile** [19] which covers a wide range of semi-supervised estimators from the literature. The package can be obtained from the Comprehensive R Archive Network (CRAN) at <http://CRAN.R-project.org/package=auxQuantile>. In Section 2, we formally define terms and provide brief introductions to the different classes of estimators with corresponding examples. The estimators described herein are not an exhaustive list and are

primarily used to highlight the differences between the types of estimators. Section 3 provides examples of how to implement some of the functions in the **auxQuantile** package on a simulated data set.

## 3.2 Auxiliary quantile estimators

Formally, let  $y_i$  denote the variable of interest and let  $U = \{1, \dots, N\}$  index all units in the population. Define  $Q_y(\alpha) = \inf \{t : F_y(t) \geq \alpha\}$  to be the  $\alpha$ -quantile of the distribution of  $y$  for some  $\alpha \in [0, 1]$ , where  $F_y(t) = N^{-1} \sum_{i \in U} I(y_i \leq t)$  is the corresponding distribution function, and  $I(\cdot)$  is the indicator function. Let  $s \subset U$  denote the sample of ‘labeled’ data of size  $n < N$ .

Let  $x_i, i \in U$ , denote the correlated auxiliary variable that is available and observed on all units in the population. The  $x_i$  for which  $i \in U/s$ , are considered ‘unlabeled’ as they do not have a corresponding  $y_i$  observation. It is assumed that if  $x$  and  $y$  are correlated then incorporating  $x$  in the estimation process will lead to better estimators of  $Q_y(\alpha)$ . Auxiliary estimators of  $Q_y(\alpha)$  can be obtained by either (1) inverting a non-decreasing auxiliary estimator of  $F_y(t)$ , call it  $\tilde{F}_y(t)$ , or (2) evaluating a auxiliary estimator of the quantile directly. We denote the first approach as CDF-based, and the second as direct-quantile based. In order to ensure that CDF-based estimators can be obtained,  $\tilde{F}_y(t)$  must be a proper CDF, i.e., must satisfy the following: (P1)  $\tilde{F}_y(t)$  is right continuous; (P2)  $\tilde{F}_y(t)$  is monotone nondecreasing; (P3)  $\lim_{t \rightarrow -\infty} \tilde{F}_y(t) = 0$ ; and (P4)  $\lim_{t \rightarrow \infty} \tilde{F}_y(t) = 1$ . If one of these do not hold, a post-estimation correction can be made. For instance, if  $\tilde{F}_y(t)$  is not monotone nondecreasing, then  $\tilde{F}_y(t)$  can be replaced with  $\tilde{F}_y^*(t)$ , where  $\tilde{F}_y^* \{y_{(1)}\} = \tilde{F}_y \{y_{(1)}\}$  and  $\tilde{F}_y^* \{y_{(i)}\} = \max \left[ \tilde{F}_y^* \{y_{(i-1)}\}, \tilde{F}_y \{y_{(i)}\} \right]$ , where  $y_{(i)}$  is the  $i$ th order statistic of  $\{y_i\}_{i \in s}$  ([21]). Alternative post-estimation corrections can be implemented if (P1), (P3), or (P4) are violated. The intuition behind CDF-based estimators is that a high-quality estimator of  $F_y(t)$  would lead to a high quality estimator of  $Q_y(\alpha)$ . However, a auxiliary estimator of  $F_y(t)$  uses auxiliary information to improve  $F_y(t)$  for all  $t$  in the domain of  $y$  when all is needed is a high-quality estimate at  $t$  near  $Q_y(\alpha)$ . Direct-quantile-based estimators attempt to take advantage of this. They estimate  $Q_y(\alpha)$  directly, by-passing the need to estimate  $F_y(t)$ .

In the sampling literature surrounding auxiliary quantile and CDF estimation, there are two competing paradigms of inference: design and model-based. In the design-based framework, all randomness arises from the sampling mechanism, call it  $p(\cdot)$ , used to generate the sample  $s$ . An estimator of  $Q_y(\alpha)$  is thus design-unbiased if the average of the estimator evaluated over all possible samples is  $Q_y(\alpha)$ . In the model-based framework, the  $(x_i, y_i)$  values are considered to be samples from a superpopulation model relating  $y$  to  $x$ . An estimator of  $Q_y(\alpha)$  is considered model-unbiased if its expectation with respect to the model is equal to  $Q_y(\alpha)$ .

Auxiliary estimators can be categorized as design-based, model-based, or model-assisted.

Design-based estimators ([60, 47, 46, 58, 34]) do not involve a superpopulation model between  $y$  and  $x$  and are thus asymptotically design-unbiased independent of a superpopulation model. However, in some cases, the asymptotic design-unbiasedness comes with increased variability.

Model-based estimators ([8, 9, 40, 14, 30, 44, 35, 9]) rely on models for  $y$  given  $x$  and heavily depend on superpopulation model assumptions. In general, they are asymptotically model-unbiased and are efficient if the assumed superpopulation model is correct but may not be consistent if the postulated model is incorrect. They are thus best utilized in settings where adequate model validation is possible.

Model-assisted ([23, 47, 46, 14, 62, 63, 51, 52, 42, 11]) estimators combine aspects of both the design- and model-based estimators. They depend on superpopulation model assumptions and gain efficiency if the assumptions are correct, but remain asymptotically design-unbiased if the model assumptions are incorrect. We further classify these as either generalized-difference or model-calibrated estimators.

### 3.2.1 Design-based estimators

Design-based estimators of  $Q_y(\alpha)$  are asymptotically design-unbiased regardless of the underlying superpopulation model for  $y$  and  $x$ . This makes them appealing in an environment where the relationship between  $y$  and  $x$  is not well understood. Denote  $\pi_i$  to be the probability of including the  $i$ th unit in the sample and let  $d_i = \pi_i^{-1}$  be the corresponding design weight. If no auxiliary information is available, a standard estimator of  $F_y(t)$  is the normalized Horvitz-Thompson estimator ([29])  $\widehat{F}_{y,HTN}(t) = (\sum_{i \in s} d_i)^{-1} \sum_{i \in s} d_i I(y_i \leq t)$ . Under simple random sampling (SRS),  $\widehat{F}_{y,HTN}$  reduces to  $\widehat{F}_{y,HTN}(t) = n^{-1} \sum_{i \in s} I(y_i \leq t)$ . Define  $\widehat{Q}_y(\alpha) = \inf \left\{ t : \widehat{F}_{y,HTN}(t) \geq \alpha \right\}$  to be the standard quantile estimator based on inverting the normalized Horvitz-Thompson estimator. Similarly, let  $\widehat{F}_{x,HTN}(t)$  denote the normalized Horvitz-Thompson estimator for the CDF of  $x$  and  $\widehat{Q}_x(\alpha)$  the estimated quantile based on inverting  $\widehat{F}_{x,HTN}(t)$ .

#### Example: Ratio estimator

In the context of estimating means or totals with auxiliary information, the ratio is one of the most popular and well-studied approaches ([12, 22]). The ratio estimator of  $F_y(t)$  is given by  $\widehat{F}_{y,Ratio}(t) = \widehat{F}_y(t)F_x(t)/\widehat{F}_x(t)$  if  $\widehat{F}_x(t) \neq 0$ , and 0 otherwise. Recall that  $F_x(t)$  is known because  $\{x_i\}_{i \in U}$  is observed. This estimator can be derived from an assumption of approximate proportionality  $F_y(t)/\widehat{F}_y(t) \approx F_x(t)/\widehat{F}_x(t)$ . See [60] for a description of the ratio estimator as a special case within a broad class of estimators defined as smooth functions of  $\widehat{F}_y(t)$ ,  $\widehat{F}_x(t)$ , and  $F_x(t)$ . Because  $F_{y,Ratio}(t)$  can take on values outside of  $[0, 1]$  and is not guaranteed to be monotone nondecreasing, it must be properly transformed before inversion takes place to estimate  $Q_y(\alpha)$ .

**Example: Direct quantile ratio estimator**

Instead of inverting a ratio estimator of the CDF, the ratio formulae can be applied to quantile estimators directly. A direct quantile based ratio estimator has the form  $\widehat{Q}_{y,Ratio}(\alpha) = \widehat{Q}_y(\alpha)Q_x(\alpha)/\widehat{Q}_x(\alpha)$  for  $\widehat{Q}_x(\alpha) \neq 0$ , where  $\widehat{Q}_x(\alpha)$  and  $\widehat{Q}_y(\alpha)$  are estimators of  $Q_x(\alpha)$  and  $Q_y(\alpha)$  respectively. Recall that  $Q_x(\alpha)$  in  $\widehat{Q}_{y,Ratio}(\alpha)$  is known. This estimator bypasses the need to invert a possibly improper CDF.

**3.2.2 Model-based estimators**

Observe that the population CDF can be decomposed as

$$F_y(t) = N^{-1} \left\{ \sum_{i \in s} I(y_i \leq t) + \sum_{i \in U \setminus s} I(y_i \leq t) \right\}.$$

In model-based estimators, the unobserved indicators functions  $I(y_i \leq t)$ ,  $i \in U \setminus s$ , are replaced by estimators of their conditional expectations  $g(x, t) = P(y \leq t|x)$ , denoted by  $\widehat{g}(x, t)$ . The resulting estimator of  $F_y(t)$  is thus

$$\widehat{F}_M(t) = N^{-1} \left\{ \sum_{i \in s} I(y_i \leq t) + \sum_{i \in U \setminus s} \widehat{g}(x_i, t) \right\}. \tag{3.1}$$

The different types of model-based estimators in (3.1) differ in the postulated model for  $g(x, t)$  and the subsequent estimator of  $g(x, t)$ .

**Example: Chambers and Dunstan estimator**

The Chambers and Dunstan [8] estimator is based on the location-scale model  $y_i = \mu(x_i) + \sigma(x_i)\epsilon_i$ ,  $i \in U$ , where  $\mu(x)$  and  $\sigma(x)$  are unknown smooth functions and  $\epsilon_1, \dots, \epsilon_N$  are i.i.d random variables with mean 0, unit variance, and unknown CDF  $G$ . Under these assumptions,  $g(x, t) = G[\{t - \mu(x)\}/\sigma(x)]$ , which can be estimated by first constructing estimators of: (i)  $\mu(x)$ , denoted by  $\widehat{\mu}(x)$ , using (non)linear least squares, (ii)  $\sigma(x)$ , denoted by  $\widehat{\sigma}(x)$ , using variance modeling, and (iii)  $G$ , denoted by  $\widehat{G}$ , using the observed residuals from the fitted model. For example, under the linear model,  $\mu(x) = \beta_0 + \beta_1 x$ , where  $\beta_0$ , and  $\beta_1$  are unknown parameters and  $\sigma(x)$  is assumed known, the Chambers and Dunstan CDF estimator is (3.1) with  $\widehat{g}(x_i, t) = \widehat{G} \left[ \left\{ t - \widehat{\beta}_0 - \widehat{\beta}_1 x_i \right\} / \sigma(x_i) \right]$ , where  $\widehat{G}(t) = n^{-1} \sum_{i \in s} I[\{y_i - \widehat{\mu}(x_i) / \sigma(x_i)\} \leq t]$  and  $\widehat{\mu}(x_i) = \widehat{\beta}_0 + \widehat{\beta}_1 x_i$ . The Chambers and Dunstan estimator of  $F_y(t)$  is a proper CDF and thus can be inverted to obtain a quantile estimator.

### Example: Kuo estimator

It may be too restrictive or unreasonable to assume a location-scale model between  $y$  and  $x$ . The Kuo estimator [35] relaxes this assumption and assumes the more general model  $P(y \leq t|x) = g(x, t)$  where  $g(x, t)$  is an unknown smooth function. An estimator of  $g(x, t)$  is  $\hat{g}(x, t) = \sum_{j \in s} \omega_j(x) I(y_j \leq t)$  where  $\omega_j(x), j \in s$ , are weights satisfying  $\omega_j(x) \geq 0$  for  $j \in s$  and  $\sum_{j \in s} \omega_j(x) = 1$ . Some examples are: (1) naive weights  $\omega_j(x) = I(|x_j - x| < \delta) / \sum_{i \in s} I(|x_i - x| < \delta)$  for  $\delta > 0$ ; (2) Nadaraya-Watson weights  $\omega_j(x) = K\{(x_j - x)/h\} / \sum_{i \in s} K\{(x_i - x)/h\}$  for some kernel function  $K(\cdot)$  and bandwidth  $h$ ; or (3) the  $k$ -nearest neighbor weights where  $\omega_j(x) = k^{-1}$  if  $x_j$  is one of the  $k$  nearest neighbors to  $x$ , and 0 otherwise. The Kuo CDF estimator is a proper CDF and thus can be inverted to estimate  $Q_y(\alpha)$ .

### 3.2.3 Model-assisted generalized difference estimators

Model-assisted estimators contain aspects of both the design and model-based estimators. They rely on a superpopulation model and also account for the sampling design. This makes them particularly efficient if the model is correctly specified while remaining asymptotically design-unbiased if the model is misspecified. The two common classes of model-assisted estimators are generalized difference and model-calibrated estimators.

Generalized difference estimators of  $F_y(t)$  have the form

$$\hat{F}_{GD}(t) = N^{-1} \left[ \sum_{i \in s} d_i I(y_i \leq t) + \sum_{i \in U} \hat{g}(x_i, t) - \sum_{i \in s} d_i \hat{g}_c(x_i, t) \right], \quad (3.2)$$

where  $\hat{g}(x_i, t)$  is a design-based estimator of  $P(y_i \leq t|x_i)$  based on a superpopulation model, and  $\hat{g}_c(x_i, t)$  is a similar estimator conditional on  $i \in s$ . Under simple random sampling,  $\hat{g}(x_i, t) = \hat{g}_c(x_i, t)$ . The idea of generalized difference estimators is to construct an estimator that is asymptotically model-unbiased under a correctly specified model while retaining asymptotic design-unbiasedness under an incorrectly specified model. These estimators are not always strictly monotone increasing nor bounded by 0 and 1, and thus must be transformed before inverting to obtain a quantile.

### Example: Rao, Kovar, and Mantel estimator

The Rao, Kovar, and Mantel estimator [47] postulates the same location-scale model as that of Chambers and Dunstan. An estimator of  $g(x, t)$  is calculated from design-based estimators of

$\mu(x)$ ,  $\sigma(x)$ , and  $G(t)$  as:

$$\hat{g}(x_i, t) = \left( \sum_{j \in s} d_j \right)^{-1} \left\{ \sum_{j \in s} d_j I[\epsilon_j \leq \{t - \hat{\mu}(x_i)\}/\hat{\sigma}(x_i)] \right\},$$

and  $\hat{g}_c(x_i, t) = \left( \sum_{j \in s} d_{j|i} \right)^{-1} \left\{ \sum_{j \in s} d_{j|i} I[\epsilon_j \leq \{t - \hat{\mu}(x_i)\}/\hat{\sigma}(x_i)] \right\}$ , where  $\epsilon_j$  are the observed residuals and  $d_{i|j} = \pi_j/\pi_{ij}$  is the inverse of the conditional probability that  $i \in s$  given  $j \in s$ . Under a design-based framework,  $\mu(x)$ ,  $\sigma(x)$ , and  $G(t)$  are estimated differently. For example, under the assumption of  $\mu(x) = \beta_1 x$  and  $\sigma(x)$  known, the design-based estimate of  $\beta_1$  is  $\hat{\beta}_1 = \{\sum_{i \in s} y_i x_i / \pi_i \sigma^2(x_i)\} \{\sum_{i \in s} x_i^2 / \pi_i \sigma^2(x_i)\}^{-1}$ , which incorporates the design weights unlike the purely model-based estimate  $\hat{\beta}_1 = \{\sum_{i \in s} y_i x_i / \sigma^2(x_i)\} \{\sum_{i \in s} x_i^2 / \sigma^2(x_i)\}^{-1}$ .

The CDF based on these estimators is asymptotically model and design-unbiased but is not a proper CDF and thus must be suitably transformed before inversion takes place.

### 3.2.4 Model-assisted calibration estimators

Another class of model-assisted estimators is model calibration ([63, 51, 27, 52]). Calibration estimators originated as estimators of population means and totals and were later modified for CDFs. First introduced by Deville and Sarndall [13], the standard estimator of the population total,  $\sum_{i \in s} d_i y_i$ , is replaced by  $\sum_{i \in s} \omega_i y_i$ , where  $\omega_i$ ,  $i \in s$ , is a set of calibrated weights chosen close to  $d_i$  while respecting the calibration equation  $\sum_{i \in s} \omega_i x_i = \sum_{i \in N} x_i$ . The calibration weights are thus perfect fits for the standard design-based estimator of  $\sum_{i \in U} y_i$  while remaining close to the original design weights  $d_i$ ,  $i \in s$ .

Model calibration estimators of  $F_y(t)$  have the form

$$\hat{F}_{MC}(t) = N^{-1} \sum_{i \in s} \omega_i(t_0) \hat{g}(x_i, t), \quad (3.3)$$

where  $\omega_i(t_0)$ ,  $i \in s$ , are weights calibrated at the point  $t_0$ , which are also made as close as possible to  $d_i$  but under the constraint  $\sum_{i \in s} \omega_i(t_0) \hat{g}(x_i, t) = \sum_{i \in U} \hat{g}(x_i, t)$ , where  $\hat{g}(x_i, t)$  are model-assisted estimates of  $I(y_i \leq t)$ . Thus if  $\hat{g}(x_i, t)$  is a good estimator of  $I(y_i \leq t)$  for  $i \in U \setminus s$  then  $\hat{F}_{MC}(t)$  is expected to perform well.

**Example: Wu and Sitter Estimator**

The Wu and Sitter [63] calibration estimator has the form (3.3) where the weights are determined by minimizing the chi-square distance function

$$\Phi_s = \sum_{i \in s} \frac{\{\omega_i(t) - d_i\}^2}{d_i q_i}, \tag{3.4}$$

subject to the calibration equations

$$\sum_{i \in s} \omega_i(t) \hat{g}(x_i, t) = \sum_{i \in U} \hat{g}(x_i, t).$$

The  $q_i$  values in (3.4) are known positive weights unrelated to  $d_i$  (typically  $q_i = 1, i \in s$ ), and the  $\hat{g}(x_i, t)$  values are estimates of  $I(y_i \leq t)$ . Because the calibration point  $t$  is not a fixed point  $t = t_0$ , different calibration weights are calculated for different estimates of  $F_y(t)$ . This makes  $\hat{F}_{MC}(t)$  not strictly monotone nondecreasing and thus for inversion to take place,  $\hat{F}_{MC}(t)$  would need to be properly transformed.

**Example: Rueda estimator**

The Rueda [54] estimator is similar to the Wu and Sitter estimator with the main difference being that calibration takes place at a fixed, possibly vector valued, point  $\mathbf{t}_0$ . It has the form (3.3) where the calibration weights are obtained by minimizing

$$\Phi_s = \sum_{i \in s} \frac{\{\omega_i(\mathbf{t}_0) - d_i\}^2}{d_i q_i}, \tag{3.5}$$

subject to the calibration equations

$$\sum_{i \in s} \omega_i(\mathbf{t}_0) \hat{g}(x_i, t_j) = \sum_{i \in U} \hat{g}(x_i, t_j), \quad j = 1, \dots, P$$

where  $q_i, i \in s$ , are known positive weights unrelated to  $d_i$  (typically  $q_i = 1, i \in s$ ),  $\hat{g}(x_i, t)$  is an estimate of  $I(y_i \leq t)$ , and  $\mathbf{t}_0 = \{t_1, \dots, t_P\}$  is a set of pre-specified points at which calibration takes place. Rueda *et al.* use  $\hat{g}(x, t) = I\{\hat{\mu}(x) \leq t\}$  and denote  $\hat{\mu}(x_i), i \in U$ , to be a set of fitted values based on an assumed model for  $\mu(x)$ . This estimator is asymptotically design-unbiased and guaranteed to be a proper CDF if  $q_i = c, i \in s$ , for some constant  $c$  and  $t_P$  is chosen large enough such that  $N^{-1} \sum_{i \in s} I\{\hat{\mu}(x_i) \leq t_P\} = 1$ . Rueda suggests choosing  $t_P = \max_{i \in U} \hat{\mu}(x_i)$ . This estimator is efficient at  $\mathbf{t}_0$  but can be inefficient at points away from  $\mathbf{t}_0$ , thus care should be taken when choosing the set of calibration points.

Table 3.1: Description of variables needed for `data`.

Variable	Support	Description
<code>Y</code>	$\mathbb{R}$	Variable of interest. Only observed in the sample.
<code>X</code>	$\mathbb{R}$	Auxiliary variable observed on all units in the population.
<code>inclusion.p</code>	$[0, 1]$	The inclusion probability for each <code>Y</code> in the sample.
<code>sample</code>	$\{0, 1\}$	Indicator for sample inclusion (1 = included, 0 otherwise).

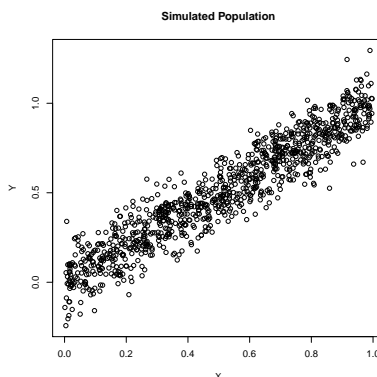


Figure 3.1: Simulated data set.

### 3.3 Using the `auxQuantile` package

#### 3.3.1 Preparing the data

Before implementing any of the CDF or quantile functions in `auxQuantile`, a data object must first be constructed with the components found in Table 3.1. Figure 3.1 contains  $N = 2000$  simulated data points from the superpopulation defined by  $y_i = x_i + \epsilon_i$ , where  $x_i$  are `Uniform(0,1)` random variables and  $\epsilon_i$  are normally distributed normal random variables with unit variance. This simulated data set is included as `ssData`. For each of the following examples, we assume  $x_i$  is available for all  $i \in U = \{1, \dots, N\}$  and  $y_i$  is available for  $i \in s$  where  $s$  is a simple random sample of size  $n = 200$ .

#### 3.3.2 Sample code for evaluating auxiliary estimators

The process of calling and executing the functions found in `auxQuantile` is similar across all of the estimators. Because of this, only a select few illustrative examples will be shown here. For more details on all of the functions available, the reader should consult [19]. Before proceeding, the `auxQuantile` must be installed and loaded. One way to do this is using the `install.packages(auxQuantile)` followed by the `library("auxQuantile")` statement.

### Example: Ratio estimator

The ratio estimator of  $Q_y(\alpha)$  can be implemented using the `GetRatio_Q` function. The inputs into the function are `alpha`, the desired quantile, and `data`, the data object described in section 3.1. The ratio estimator can be evaluated in the following way for  $\alpha = .05$  and  $\alpha = .95$ :

```
> GetRatio_Q(alpha=.05, data=ssData)
```

```
[1] 0.0426109
```

```
> GetRatio_Q(alpha=.95, data=ssData)
```

```
[1] 0.9893489.
```

### Example: Chambers and Dunstan estimator

The Chambers and Dunstan [8] estimator is slightly more involved than the ratio estimator used in `GetRatio_Q`. The assumed relationship between  $y$  and  $x$  is

$$y = \mu(x) + \sigma(x_i)\epsilon,$$

which requires assumptions regarding  $\mu(x)$  and  $\sigma(x)$ . The inputs required are displayed in Table 3.2. If  $\mu(x)$  is assumed to have a linear parametric form, namely  $\mu(x) = \beta_0 + \sum_{i=1}^p \beta_i x_i$ , then `parametric` should be set to `TRUE` and the syntax in the `model` statement should be equivalent to that used by the `lm` function. Alternatively, if  $\mu(x)$  is assumed to have a nonparametric form (`parametric = FALSE`), namely  $\mu(x) = \sum_{j \in s} \omega_j(x) y_j$ , then a  $n \times N$  matrix of  $\omega$  weights (one  $n \times 1$  vector of weights to estimate each  $\mu(x_i)$ , for  $i \in U$ ) should be provided in the `omega.mu` statement. Otherwise, `omega.mu` can be set to `NULL` and the weights will be estimated using one of methods specified in the `np.weights.mu` statement.

If  $\sigma(x)$  is assumed constant, any positive number can be used in the `sigma` statement, otherwise, a set of weights must be provided or estimated similar to the nonparametric estimation of  $\mu(x)$ . A third bias-robust term can also be included in the model and is estimated similar to  $\mu(x)$  and  $\sigma(x)$ . Details of this form of the Chambers and Dunstan estimator are outlined in [9] Chambers et al. (1993) and [20]. Each of the tuning parameters (`b.mu`, `b.sig`, `b.bias`) can be prespecified or estimated from the data using a general rule of thumb proposed by [59].

The following code and output demonstrates how to evaluate the `GetModelCD_Q` function at  $\alpha = .95$  assuming  $\mu(x) = \beta_0 + \beta_1 x$  and  $\sigma(x) = 1$ :

Table 3.2: Description of variables needed for `GetModelCD.Q`.

Variable	Description
<code>alpha</code>	The desired quantile
<code>data</code>	The data object
<code>parametric</code>	<b>true</b> if parametric model is assumed for $\mu(x)$ <b>false</b> if a nonparametric model is assumed for $\mu(x)$
<code>model</code>	If <code>parametric = TRUE</code> , the formula for $\mu(x)$ used by the <code>lm</code> function.
<code>omega.mu</code>	The $n \times N$ matrix of $\omega_i(x_j)$ weights for $i \in s, j \in U$ , needed to estimate $\mu(x)$ nonparametrically. If <code>NULL</code> (default), the weights are estimated.
<code>omega.sig</code>	The $n \times N$ matrix of $\omega_i(x_j)$ weights for $i \in s, j \in U$ , needed to estimate $\sigma(x)$ nonparametrically. If <code>NULL</code> (default), the weights are estimated.
<code>omega.bias</code>	The $n \times N$ matrix of $\omega_i(x_j)$ weights for $i \in s, j \in U$ , needed to estimate the bias-robust term nonparametrically. If <code>NULL</code> (default), the weights are estimated.
<code>np.weights.mu</code>	This determines the weighting method used to compute the nonparametric weights if <code>omega.mu = NULL</code> (default). The choices are local linear regression ( <code>LLR</code> ), kernel regression using a normal density ( <code>K-NORM</code> ), Nadaraya Watson ( <code>NW</code> ), or k-nearest neighbors ( <code>KNN</code> ).
<code>np.weights.sig</code>	This determines the weighting method used to compute the nonparametric weights if <code>omega.sig = NULL</code> (default). For choices, see <code>np.weights.mu</code> .
<code>np.weights.bias</code>	This determines the weighting method used to compute the nonparametric weights if <code>omega.bias = NULL</code> (default). For choices, see <code>np.weights.mu</code> .
<code>b.mu</code>	If <code>omega.mu = NULL</code> (default), this is the bandwidth used when estimating the nonparametric weights. If <code>b.mu = NULL</code> (default), <code>b.mu</code> is estimated.
<code>b.sig</code>	If <code>omega.sig = NULL</code> (default), this is the bandwidth used when estimating the nonparametric weights. If <code>b.sig = NULL</code> (default), <code>b.sig</code> is estimated.
<code>b.bias</code>	If <code>omega.bias = NULL</code> (default), this is the bandwidth used when estimating the nonparametric weights. If <code>b.bias = NULL</code> (default), <code>b.sig</code> is estimated.

Table 3.3: Description of variables needed for `GetModelK_Q`.

Variable	Description
<code>alpha</code>	The desired quantile
<code>data</code>	The data object
<code>omega.cdf</code>	The $n \times N$ matrix of $\omega_i(x_j)$ , $i \in s, j \in U$ , needed to estimate $g(x, t)$ nonparametrically. If <code>NULL</code> (default), the weights are estimated.
<code>np.weights.cdf</code>	If <code>omega.cdf = NULL</code> (default), this determines the weighting method used to compute the nonparametric weights. The choices are local linear regression (LLR), kernel regression using a normal density (K-NORM), Nadaraya Watson (NW), or k-nearest neighbors (KNN).
<code>b.cdf</code>	If <code>omega.cdf = NULL</code> (default), this is the bandwidth used when estimating the nonparametric weights. If <code>b.cdf = NULL</code> (default), <code>b.mu</code> is estimated.

```
> GetModelCD_Q(alpha=.95, data=ssData, parametric=TRUE, model=Y~X, sigma=1)
```

```
[1] 2.223702.
```

If  $\mu(x)$  and  $\sigma(x)$  are assumed to be unknown and estimated from the observed data using local-linear regression weights, the following code can be ran:

```
> GetModelCD_Q(alpha=.95, data=ssData, parametric=FALSE, np.weights.mu = 'LLR')
```

```
[1] 2.222845.
```

The Chambers and Dunstan estimator of  $F_y(t)$  can be evaluated directly using `GetModelCD_CDF` function using the same syntax as `GetModelCD_Q` with the only difference being that instead of specifying the  $\alpha$  at which to evaluate the quantile, the value(s) of  $t$  must be provided in the `t` statement. For example, to evaluate the CDF estimator at  $t = 0$  and  $t = 2.5$ , the following code can be ran:

```
> GetModelCD_CDF(t=c(0,2.5), data=ssData, parametric=FALSE)
```

```
[1] 2.222845 2.643643.
```

### Example: Kuo estimator

Instead of assuming an additive model between  $y$  and  $x$ , the Kuo [35] estimator of  $F_y(t)$  assumes the more general model  $P(y \leq t|x) = g(x, t)$  with corresponding nonparametric estimator  $\hat{g}(x, t) = \sum_{j \in s} \omega_j(x) I(y_j \leq t)$ . The `GetModelK_CDF` function evaluates this CDF estimator and the `GetModelK_Q` function evaluates the corresponding quantile estimator. The inputs for `GetModelK_Q` are found in Table 3.3. The following code demonstrates how to execute the `GetModelK_Q` function using the Nadaraya Watson weighting scheme:

```
> GetModelK_Q(alpha=.05, data=ssData, omega.cdf=NULL, np.weights.cdf='NW',
              b.cdf=NULL)
```

```
[1] -1.113045.
```

### Example: Rueda estimator

The Rueda [54] estimator can be evaluated using the `GetModelCalR_Q` function. Before evaluating the function, a model for  $\mu(x)$  must be specified as well as the point(s) of calibration  $\mathbf{t}_0 = \{t_1, \dots, t_P\}$ . The model for  $\mu(x)$  is specified in the same way as `GetModelCD_Q`. The  $\mathbf{t}_0$  points are chosen based on inputted quantiles of the  $\hat{\mu}(x)$  fitted values from `t.quant`. For instance, `t.quant = c(.25, .75)` would calibrate on the 25th and 75th quantiles of  $\hat{\mu}(x_i)$ ,  $i \in U$ . The following code and output demonstrates how to execute the `GetModelCalR_Q` function assuming a simple linear model for  $\mu(x)$ :

```
> GetModelCal_Q(alpha=.9, data=ssData, parametric=TRUE, model=Y~X)
```

```
[1] 2.113045.
```

## 3.4 Conclusion

We have demonstrated how to compute estimators of the population quantile and CDF using the `auxQuantile` package in **R**. The `auxQuantile` package contains various estimators of the population quantile or CDF which incorporate a correlated auxiliary variable. For the model-based and model-assisted estimators, the functions in `auxQuantile` encourage proper model-selection and diagnostic methods.

# Chapter 4

## Auxiliary Bootstrap Methods

### 4.1 Introduction

In this chapter, we describe the auxiliary bootstrap approach. In section 4.2, we formalize the estimation problem and derive the auxiliary bootstrap estimator. In Section 4.4, we examine the finite sample performance of the proposed estimators on a series of simulation experiments and an application to a geospatial weather model. We offer concluding remarks in Section 4.5.

### 4.2 Auxiliary bootstrap

Let  $\mathbf{X} \in \mathcal{X}$  be a random variable with fixed but unknown distribution  $P$ . We assume that the available data are  $\{\mathbf{X}_i\}_{i=1}^n$  which comprise  $n$  independent replicates of  $\mathbf{X}$ . Let  $\mathbb{P}_n$  denote the empirical measure, i.e.,  $\mathbb{P}_n \triangleq n^{-1} \sum_{i=1}^n \delta_{\mathbf{X}_i}$  where  $\delta_u$  is the Dirac measure at  $u$ . Define the bootstrap empirical measure  $\widehat{\mathbb{P}}_n^* \triangleq n^{-1} \sum_{i=1}^n M_{n,i} \delta_{\mathbf{X}_i}$ , where  $(M_{n,1}, \dots, M_{n,n})$  is a multinomial random vector that is independent of the observed data and is indexed by  $n$  parameters each equal to  $1/n$ . Given a functional of interest,  $F_n = F(\mathbb{P}_n, P) \in \mathbb{R}$ , let  $\widehat{F}_n^* = F(\widehat{\mathbb{P}}_n^*, \mathbb{P}_n)$  denote its bootstrap analog.

Let  $P_M$  denote probability taken with respect to the multinomial weights. Define  $\lambda_n$  to be some attribute of the sampling distribution of  $F_n$  such as the quantile, mean, or variance. We denote  $\widehat{\lambda}_n$  to be the bootstrap estimator of  $\lambda_n$ , which can be approximated by  $\widehat{\lambda}_n^*(B) = \lambda(F_n^{*(1)}, F_n^{*(2)}, \dots, F_n^{*(B)})$  where  $\widehat{F}_n^{*(1)}, \dots, \widehat{F}_n^{*(B)}$  are bootstrap estimators constructed across  $B$  independent and identically distributed draws of the multinomial weights. The number of resamples  $B$  is often taken to be large to reduce Monte Carlo error. Let  $\widehat{\lambda}_n^*(\infty)$  denote the ‘ideal’ estimator [2] where an infinite number of Monte Carlo resamples can be obtained. In practice, this of course is not feasible and a smaller number of resamples must be used instead, potentially leading to a low quality approximation.

Suppose that there exists a surrogate  $S_n = S(\mathbb{P}_n, P)$  for  $F_n$  whose bootstrap analog,  $\widehat{S}_n^*$ , is computationally inexpensive relative to  $\widehat{F}_n^*$ ; examples of such surrogates are given in Section 3 (see also [6]). We assume that one will generate bootstrap resamples  $\left\{ \left( \widehat{F}_n^{*(m)}, \widehat{S}_n^{*(m)} \right) \right\}_{m=1}^M$  and  $\left\{ \widehat{S}_n^{*(M+k)} \right\}_{k=1}^K$ ; methods for choosing  $M$  and  $K$  will be discussed in the next section. A natural approach to incorporating the surrogate resamples into an estimator for  $\lambda_n$  is through a working model that links the estimator of interest to the surrogate. We consider parametric location-scale models of the form  $F_n = g(S_n; \beta^*) + \tau^* Z / \sqrt{n}$ , where  $\beta^* \in \mathbb{R}^p$  and  $\tau^* > 0$  are unknown parameters,  $g$  is fixed and known, and  $Z$  is an independent error which has mean zero and variance one. In our working model, the distribution of  $Z$  will be assumed to be standard normal; however, this not required as this distribution could be estimated from the observed data (see below). Define

$$\widehat{\beta}_{n,M} = \arg \min_{\beta \in \mathbb{R}^p} \sum_{m=1}^M \left\{ \widehat{F}_n^{*(m)} - g \left( \widehat{S}_n^{*(m)}; \beta \right) \right\}^2,$$

and subsequently

$$\widehat{\tau}_{n,M}^2 = \frac{n}{M} \sum_{m=1}^M \left\{ \widehat{F}_n^{*(m)} - g \left( \widehat{S}_n^{*(m)}; \widehat{\beta}_{n,M} \right) \right\}^2.$$

If one were unwilling to assume that  $Z$  were normally distributed, then the standardized residuals  $\sqrt{n} \left\{ \widehat{F}_n^{*(m)} - g \left( \widehat{S}_n^{*(m)}; \widehat{\beta}_{n,M} \right) \right\} / \widehat{\tau}_{n,M}$ ,  $m = 1, \dots, M$ , could be used to construct an estimator of its distribution; e.g., one could use the empirical distribution of these standardized residuals.

In Sections 4.2.1, 4.2.2, and 4.2.3, we define  $\lambda_n, \widehat{\lambda}_n, \widehat{\lambda}_n^*(B)$ , and the auxiliary estimator  $\widehat{\lambda}_n^{aux}(M, K)$ , for estimating the quantile, mean, and variance, respectively, of the sampling distribution of  $F_n$ .

#### 4.2.1 Auxiliary quantile estimator

Let  $\lambda_n = (Q_n)^{-1}(\alpha)$  where  $Q_n(t) = P(F_n \leq t)$ . The bootstrap estimator of  $(Q_n)^{-1}(\alpha)$  is  $\widehat{\lambda}_n^* = \left( \widehat{Q}_n^* \right)^{-1}(\alpha)$  where  $\widehat{Q}_n(t) \triangleq P_M \left( \widehat{F}_n^* \leq t \right)$  which can be estimated via Monte Carlo by  $\widehat{\lambda}_n^*(B) = \left( \widehat{Q}_n^*(B) \right)^{-1}(\alpha)$  where

$$\widehat{Q}_n^*(t; B) = \frac{1}{B} \sum_{b=1}^B 1_{\widehat{F}_n^{*(b)} \leq t}, \quad (4.1)$$

Define  $R_n(t, s_n) = P(F_n \leq t | S_n = s_n)$  then  $R_n(t, S_n)$  is unbiased for  $Q_n(t)$ . Let  $\Phi$  denote the cumulative distribution function for a standard normal random variable, then under the

postulated model, it can be seen that  $Q_n(t) = \mathbb{E}R_n^*(t, S_n) = \Phi[\sqrt{n}\{t - g(S_n; \beta^*)\} / \tau^*]$ . The auxiliary quantile bootstrap estimator is  $\widehat{\lambda}_n^{\text{aux}}(M, K) = \left(\widehat{Q}_n^{\text{aux}}\right)^{-1}(\alpha; M, K)$  where

$$\widehat{Q}_n^{\text{aux}}(t; M, K) = \frac{1}{2M + K} \sum_{m=1}^M 1_{\widehat{F}_n^{*(m)} \leq t} + \frac{1}{2M + K} \sum_{k=1}^{K+M} \Phi \left[ \frac{\sqrt{n} \left\{ t - g \left( \widehat{S}_n^{*(k)}; \widehat{\beta}_{n,M} \right) \right\}}{\widehat{\tau}_{n,M}} \right],$$

where the first term is the bootstrap empirical distribution,  $\widehat{Q}_n^*(t; M)$ , and the second term is a plug-in estimator of  $\mathbb{E}R_n^*(t, \widehat{S}_n^*)$ . This estimator is analogous to the Chambers-Dunstan cdf estimator used in survey sampling [8, 47]. It can be shown that this estimator is a proper cumulative distribution function and thus can be inverted.

### 4.2.2 Auxiliary mean estimator

Let  $\lambda_n = \mu_n = \mathbb{E}[F_n]$ . The bootstrap estimator is denoted as  $\widehat{\lambda}_n = \widehat{\mu}_n = \mathbb{E}_M[\widehat{F}_n^*]$ , where  $\mathbb{E}_M$  denotes expectation with respect to the multinomial weights. Let  $\widehat{\lambda}_n^*(B)$  be the Monte Carlo based bootstrap estimator of  $\mu$ , defined as

$$\widehat{\lambda}_n^*(B) = \widehat{\mu}_n^*(B) = \frac{1}{B} \sum_{b=1}^B \widehat{F}_n^{*(b)}. \quad (4.2)$$

Recall, that by the law of total expectation,  $\mathbb{E}[F_n] = \mathbb{E}[\mathbb{E}\{F_n | S_n\}] = \mathbb{E}[g(S_n; \beta^*)]$ . We denote the auxiliary estimator of  $\mu$  as

$$\widehat{\lambda}_n^{\text{aux}}(M, K) = \widehat{\mu}_n^{\text{aux}}(M, K) = \frac{1}{M + K} \sum_{i=1}^{M+K} \left[ g(\widehat{S}_n^{*(i)}; \widehat{\beta}_{n,M}) \right], \quad (4.3)$$

### 4.2.3 Auxiliary variance estimator

Let  $\lambda_n = \sigma_n^2 = \text{Var}[F_n]$ . The bootstrap estimator is denoted as  $\widehat{\lambda}_n = \widehat{\sigma}_n^2 = \text{Var}_M[\widehat{F}_n^*]$ , where  $\text{Var}_M$  denotes the variance with respect to the multinomial weights. Let  $\widehat{\lambda}_n^*(B)$  be the Monte Carlo based bootstrap estimator of  $\sigma^2$ , defined as

$$\widehat{\lambda}_n^*(B) = \widehat{\sigma}_n^{2*}(B) = \frac{1}{B-1} \sum_{b=1}^B \left\{ \widehat{F}_n^{*(b)} - \widehat{\mu}_n^*(B) \right\}^2 \quad (4.4)$$

Recall, that by the law of total variance,  $\text{Var}[F_n] = \mathbb{E}[\text{Var}\{F_n|S_n\}] + \text{Var}[\mathbb{E}\{F_n|S_n\}] = \tau^{2^*}/n + \text{Var}[g(S_n; \beta^*)]$ . We denote the auxiliary estimator of  $\sigma^2$  as

$$\widehat{\lambda}_n^{aux}(M, K) = \widehat{\sigma}_n^{2aux}(M, K) = \widehat{\tau}_{n,M}^2/n + \frac{1}{M+K} \sum_{i=1}^{M+K} \left[ g(\widehat{S}_n^{*(i)}; \widehat{\beta}_{n,M}) - \widehat{\mu}_n^{aux}(M, K) \right]^2. \quad (4.5)$$

#### 4.2.4 Choosing the number of surrogates

The choice of  $M$  and  $K$  dictate both solution quality and computation time. In this section, we provide a simple heuristic method for choosing  $M$  and  $K$  so as to ensure high-quality estimation of  $\lambda_n$ . All derivations are conditional on the observed data.

Define  $\Gamma(M, K)$  to be the variance of  $\widehat{\lambda}_n^{aux}(M, K)$  conditional on  $\mathbb{P}_n$ . Suppose that the cost of computing an instance of  $\widehat{F}_n^*$  is  $c_F$  whereas the cost of computing an instance of  $\widehat{S}_n^*$  is  $c_S$ . Given a budget  $L \in \mathbb{R}_+$ , we define the optimal allocation, say  $\{M^{\text{opt}}(L), K^{\text{opt}}(L)\}$ , as any solution to

$$\begin{aligned} \min_{M, K \in \mathbb{Z}_+} \quad & \Gamma(M, K) \\ \text{s.t.} \quad & (c_F + c_S)M + c_S K \leq L. \end{aligned} \quad (4.6)$$

While the above expression is a non-linear integer program, it can be solved quickly through brute force by considering only values of  $(M, K)$  on the boundary of the constraint  $(c_F + c_S)M + c_S K \leq L$ , i.e., by considering only pairs  $(M, K)$  such that  $(M+1, K)$  and  $(M, K+1)$  would both violate the constraint. The set of boundary points is

$$\{(M, \lfloor c_S^{-1}[L - (c_F + c_S)M] \rfloor) : M \in \mathbb{Z}_+, 0 \leq M \leq \lfloor (c_F + c_S)^{-1}L \rfloor\}.$$

To construct an estimator of  $\{M^{\text{opt}}(L), K^{\text{opt}}(L)\}$  we: (S1) compute an initial set of bootstrap pairs  $\mathcal{D}_{M_0} = \left\{ \left( \widehat{F}_n^{*(m)}, \widehat{S}_n^{*(m)} \right) \right\}_{m=1}^{M_0}$ ; (S2) construct an estimator,  $\widehat{\Gamma}_{n, M_0}(M, K)$  of  $\Gamma(M, K)$  using the bootstrap applied to  $\mathcal{D}_{M_0}$ ; and (S3) plug  $\widehat{\Gamma}_{n, M_0}(M, K)$  into (4.6) and solve to obtain the estimators  $\left\{ \widehat{M}_{n, M_0}(L), \widehat{K}_{n, M_0}(L) \right\}$  of  $\{M^{\text{opt}}(L), K^{\text{opt}}(L)\}$ . A key point is that the bootstrap in (S2) does not require recomputing the functional  $F_n$  and therefore can be done efficiently.

Let  $G$  denote the conditional distribution of  $(\widehat{F}_n^*, \widehat{S}_n^*)$  given  $\mathbb{P}_n$ , then it can be seen that  $\Gamma(M, K)$  is completely determined by  $(G, M, K)$ . Let  $\widehat{G}_{n, M_0}$  denote an estimator of the conditional distribution of  $(\widehat{F}_n^*, \widehat{S}_n^*)$  given  $\mathbb{P}_n$  constructed from  $\mathcal{D}_{M_0}$ ; in our applications, we use a Gaussian copula though other estimators can be used [56]. Define  $\widehat{\Gamma}_{n, M_0}(M, K)$  to be the estimator of  $\Gamma(M, K)$  constructed via Monte Carlo approximation using  $\widehat{G}_{n, M_0}$  and let  $\left\{ \widehat{M}_{n, M_0}(L), \widehat{K}_{n, M_0}(L) \right\}$  denote the subsequent estimated optimal allocation derived from (4.6).

The estimated efficient frontier of  $\lambda_n$  is therefore

$$\widehat{\mathcal{F}}_{n,M_0} = \left\{ \widehat{\Gamma}_{n,M_0} \left[ \widehat{M}_{n,M_0}(L), \widehat{K}_{n,M_0}(L) \right] : L \in \mathbb{R}_+ \right\}.$$

The estimated efficient frontier can be used to visualize the trade-off between computational expenditure and solution quality or to solve for a desired level of precision.

For example, one might wish to estimate the minimal allocation for which the variance of the auxiliary quantile estimator is equal to the variance of the bootstrap empirical quantile computed using a pre-specified number of bootstrap samples. Let  $q_n^*(t)$  denote the conditional density of  $\widehat{F}_n^*$ , the then asymptotic variance of the  $\alpha \times 100$  empirical quantile is  $\sigma(\alpha) = \left[ q_n \left\{ \widehat{Q}_n^{-1}(\alpha) \right\}^2 \right]^{-1} \alpha(1-\alpha)$ . Approximating the density  $q_n^*(t)$  with

$$\begin{aligned} \widehat{q}_{n,M_0}(t) &= \frac{1}{M_0} \sum_{k=1}^{M_0} \frac{d}{dt} \Phi \left[ \frac{\sqrt{n} \left\{ t - g \left( \widehat{S}_n^{*(k)}; \widehat{\beta}_{n,M_0} \right) \right\}}{\widehat{\tau}_{n,M_0}} \right] \\ &= \frac{\sqrt{n}}{\widehat{\tau}_{n,M_0} M_0} \sum_{k=1}^{M_0} \phi \left[ \frac{\left\{ t - g \left( \widehat{S}_n^{*(k)}; \widehat{\beta}_{n,M_0} \right) \right\}}{\widehat{\tau}_{n,M_0}} \right], \end{aligned}$$

and using the approximated quantile  $\left( \widehat{Q}_n^{aux} \right)^{-1}(\alpha; M_0, 0)$  yields an estimated variance of the bootstrap empirical quantile estimator based on  $B$  resamples of

$$\widehat{J}_{n,M_0}(B) = \left( B \widehat{q}_{n,M_0}^* \left[ \left\{ \widehat{Q}_n^{aux} \right\}^{-1}(\alpha; M_0, 0) \right]^2 \right)^{-1} \alpha(1-\alpha).$$

When estimating  $\mu_n$  or  $\sigma_n^2$ ,  $\widehat{J}_{n,M_0}(B) = B^{-1} \widehat{\sigma}_n^{2aux}(M_0, 0)$  or  $\widehat{J}_{n,M_0}(B) = 2(B-1)^{-1} \widehat{\sigma}_n^{2aux}(M_0, 0)^2$  can be used instead respectively. Let  $\widehat{L}_{n,M_0}(B)$  denote the smallest cost such that

$$\widehat{\Gamma}_{n,M_0} \left[ \widehat{M}_{n,M_0}(L), \widehat{K}_{n,M_0}(L) \right]$$

is smaller than  $\widehat{J}_{n,M_0}(B)$ . In our simulations, we search over a finite grid of  $(M, K)$  values that satisfy a budget less than or equal to  $B$ . The corresponding estimated optimal allocation using the surrogate is

$$\left[ \widehat{M}_{n,M_0} \left\{ \widehat{L}_{n,M_0}(B) \right\}, \widehat{K}_{n,M_0} \left\{ \widehat{L}_{n,M_0}(B) \right\} \right].$$

The preceding approach can be used to ensure that the auxiliary bootstrap estimator does not result in reduced solution quality relative to the standard bootstrap estimator.

In some cases, one may wish to determine values of  $M$  and  $K$  for multiple quantiles simultaneously. Let  $\Gamma(\alpha; M, K)$  denote the variance of  $(Q^{aux})^{-1}(\alpha, M, K)$ . Given  $\alpha_1, \alpha_2 \in (0, 1)$ , one can compute the optimal allocation for the  $\alpha_1 \times 100$  and  $\alpha_2 \times 100$  quantiles by solving

$$\begin{aligned} \min_{M, K \in \mathbb{Z}_+} \quad & \max \left[ \widehat{\Gamma}_{n, M_0}(\alpha_1; M, K), \widehat{\Gamma}_{n, M_0}(\alpha_2; M, K) \right] \\ \text{s.t.} \quad & (c_F + c_S)M + c_S K \leq L, \end{aligned} \quad (4.7)$$

where we have minimized the maximum estimated variances across the two quantile of interest; alternatively, one could minimize the average of the two estimated variances.

#### 4.2.5 Choosing which surrogates to label

Suppose that one has computed  $\mathcal{D}_0 = \left\{ \left( \widehat{F}_n^{*(m)}, \widehat{S}_n^{*(m)} \right) \right\}_{m=1}^{M_0}$  and that it is desired to compute an additional  $M \in \mathbb{Z}_+$  replicates of  $(\widehat{F}_n^*, \widehat{S}_n^*)$  and an additional  $K \in \mathbb{Z}_+$  replicates of  $\widehat{S}_n^*$ . One can first compute  $(M + K)$  replicates of  $\widehat{S}_n^*$ , say  $\left\{ \widehat{S}_n^{*(M_0+b)} \right\}_{b=1}^{M+K}$ , and subsequently apply classical methods from experimental design to select  $M$  of these points to ‘label’ with their corresponding instances of  $\widehat{F}_n^*$  so as to minimize some design criterion. A simple criterion, one that we employ in our simulations, is to minimize the estimated sampling variability of the  $\widehat{\beta}_{n, M+M_0}$ .

Let  $\{j_1, \dots, j_M\} \subseteq \{1, \dots, M + K\}$  denote the replicates to be labeled. A first-order linear approximation of  $g(\widehat{F}_n^*; \beta^*)$  shows that the asymptotic variance of  $\widehat{\beta}_{n, M+M_0}$  under this labeling is proportional to

$$\begin{aligned} \sigma(j_1, \dots, j_M; \mathcal{D}_0) \quad & \propto \quad \text{tr} \left\{ \sum_{\ell=1}^{M_0} \nabla g(\widehat{S}_n^{*(\ell)}; \widehat{\beta}_{n, M_0}) \nabla g^\top(\widehat{S}_n^{*(\ell)}; \widehat{\beta}_{n, M_0}) \right. \\ & \left. + \sum_{\ell=1}^K \nabla g(\widehat{S}_n^{*(M_0+j_\ell)}; \widehat{\beta}_{n, M_0}) \nabla g^\top(\widehat{S}_n^{*(M_0+j_\ell)}; \widehat{\beta}_{n, M_0}) \right\}^{-1}, \end{aligned}$$

where  $\widehat{\beta}_{n, M_0}$  is a functional of  $\mathcal{D}_0$ . Thus, one can choose  $j_1, \dots, j_M$  to be the minimizers of  $\sigma(j_1, \dots, j_M; \mathcal{D}_0)$ . Computing this minimization may be computationally burdensome for large values of  $M$  or  $K$  and in such cases heuristic numerical search methods can be used [43]. In our implementation, we used coordinate descent [3] with a fixed number of coordinate updates to limit computation time.

In settings where the joint optimization of  $\sigma(j_1, \dots, j_M; \mathcal{D}_0)$  is too computationally burdensome, one can select indices to label using a stepwise Gauss-Seidel procedure as follows. Define  $\widehat{j}_{1, \mathcal{D}_0} = \arg \min_{j_1 \in \{1, \dots, M+K\}} \sigma(j_1; \mathcal{D}_0)$  and subsequently  $\mathcal{D}_1 = \mathcal{D}_0 \cup \left\{ \left( \widehat{F}_n^{*(\widehat{j}_1)}, \widehat{S}_n^{*(\widehat{j}_1)} \right) \right\}$ ; then recursively for  $v = 2, \dots, K$  define  $\widehat{j}_v = \arg \min_{j_v \in \{1, \dots, K+M\} \setminus \{\widehat{j}_1, \dots, \widehat{j}_{v-1}\}} \sigma(\widehat{j}_1, \dots, \widehat{j}_{v-1}, j_v; \mathcal{D}_{v-1})$  and subsequently  $\mathcal{D}_v = \mathcal{D}_{v-1} \cup \left\{ \left( \widehat{F}_n^{*(\widehat{j}_v)}, \widehat{S}_n^{*(\widehat{j}_v)} \right) \right\}$ .

### 4.3 Proof of concept simulations

In this section we conduct a proof of concept simulation experiment to compare the auxiliary estimators against their standard counterparts. We denote  $Y$  to be the variable of interest and  $X$  to be a noisy, computationally cheaper, surrogate of  $Y$ . In terms of the bootstrap,  $Y$  is akin to the expensive bootstrap functional  $\widehat{F}_n^*$ , and  $X$  is similar to  $\widehat{S}_n^*$ . Let  $\lambda$  denote a characteristic of the distribution of  $Y$ . For this simulation experiment, we limit  $\lambda$  to be the quantile, mean, or variance of  $Y$ . When estimating the quantile, we consider  $\alpha = \{.8, .9, .95, .99\}$ .

We are interested in estimating  $\lambda$  from the sample  $(Y_1, X_1), (Y_2, X_2), \dots, (Y_M, X_M)$  along with the additional sample  $X_{M+1}, \dots, X_{M+K}$ . Let  $\widehat{\lambda}(B)$  be the standard estimator of  $\lambda$  and let  $\widehat{\lambda}^{aux}(M, K)$  be the corresponding auxiliary estimator. Let  $F_X$  and  $F_Y$  denote the marginal distributions of  $Y$  and  $X$ . Let  $F_{Y|X}$  denote the conditional distribution of  $Y$  given  $X$ . We assume that the cost of drawing  $Y$  from either  $F_Y$  or  $F_{Y|X}$  is  $C_Y$  and the cost of drawing  $X$  from  $F_X$  is  $C_X$ . We also assume  $C_X \ll C_Y$ .

#### 4.3.1 Model

We assume the joint distribution of  $X$  and  $Y$  to be

$$\begin{pmatrix} X \\ Y \end{pmatrix} \sim N \left[ \begin{pmatrix} \mu_X \\ \mu_Y \end{pmatrix}, \begin{pmatrix} \sigma_X^2 & \rho \\ \rho & \sigma_Y^2 \end{pmatrix} \right],$$

where  $\mu_X$  and  $\sigma_X^2$  are the mean and variance of  $X$  and  $\rho$  is the correlation between  $X$  and  $Y$ . Thus marginally,  $Y \sim N(\mu_Y, \sigma_Y^2)$  and  $X \sim N(\mu_X, \sigma_X^2)$ . We also assume that given  $X$ , the conditional distribution of  $Y$  is  $N(\beta_0 + \beta_1 X, \sigma_{Y|X}^2)$ , where  $\beta_0$  and  $\beta_1$  are linear model coefficients and  $\sigma_{Y|X}^2$  is the conditional variance of  $Y$  given  $X$ , assumed constant for all  $X$ . We define  $\Omega = \{\mu_Y, \sigma_Y^2, \mu_X, \sigma_X^2, \beta_0, \beta_1, \sigma_{Y|X}^2, \rho\}$ .

For all simulations, we assume  $\mu_Y = \mu_X = 0$  and  $\sigma_Y^2 = \sigma_X^2 = 1$ . We set  $\rho$  to be one of the values in  $\{.6, .7, .8, .9, .95, .99\}$ , thus allowing us to assess the impact of the correlation between  $X$  and  $Y$  on the precision of the auxiliary estimators.

#### 4.3.2 Simulation design

We fix  $C_Y = 1$  and consider two values of  $C_X$ ,  $.1$ , and  $.05$  (denoting  $X$  being either 10 or 20 times less expensive to sample than  $Y$ ). We consider three budgets  $L = \{50, 100, 250\}$ . For each unique combination of  $\rho$ ,  $L$  and  $C_X$ , we run  $M = 500$  Monte Carlo simulations. For each simulation, we estimate  $\widehat{\lambda}(L)$ ,  $\widehat{\lambda}^{aux}(L) = \widehat{\lambda}^{aux}(\widehat{M}_{M_0}(L), \widehat{K}_{M_0}(L))$ , and  $\widehat{\lambda}^{aux}(\widehat{L}) = \widehat{\lambda}^{aux}(\widehat{M}_{M_0}(\widehat{L}), \widehat{K}_{M_0}(\widehat{L}))$ . We measure the performance of each estimator by its mean squared error (MSE).

#### 4.3.3 Results

The results of the simulations are found in the following sections. Only results for  $L = 500$  and  $C_x = .05$  are included here (remaining results can be found in the Appendix) as the conclusions are similar, yet slightly less pronounced, for  $L = 100$  and  $C_x = .1$ . The results consistently show that the auxiliary

estimators perform better than or not significantly different from their standard counterparts and perform particularly strong when the correlation between  $X$  and  $Y$  is high.

## Quantile

The results of simulations for estimating  $(Q)^{-1}(\alpha)$  for a budget of  $L = 500$  and  $C_x = .05$  are in Figure 4.1. The results for  $L = 100$  and  $C_x = .1$  are similar and can be found in the Appendix. In all cases,  $\lambda^{aux}(L)$  had a significantly lower MSE compared to  $\lambda^*(L)$ , particularly so as  $\rho$  approached 1. Additionally, in all scenarios, the MSE of  $\lambda^{aux}(\widehat{L})$  was always either significantly less than or not significantly different from that of  $\lambda^*(L)$  while using a much smaller budget. This suggests that the auxiliary quantile approach can lead to more precise estimators under a fixed budget, or to less expensive estimators with equal, or better, precision. The plots also show that estimating  $(Q)^{-1}(.99)$  is more difficult than the other quantiles and benefits the most from the auxiliary bootstrap approach.

## Mean

The results of simulations for estimating  $\mu$  for a budget of  $L = 500$  and  $C_x = .05$  are in Figure 4.2. The results for  $L = 100$  and  $C_x = .1$  are similar and can be found in the Appendix. In all cases,  $\lambda^{aux}(L)$  had a significantly lower MSE compared to  $\lambda^*(L)$ . Additionally, in all scenarios, the MSE of  $\lambda^{aux}(\widehat{L})$  was always either significantly less than or not significantly different from that of  $\lambda^*(L)$  and used a much smaller budget. This suggests that the auxiliary approach can lead to more precise estimators under a fixed budget, or to less expensive estimators with equal, or better, precision.

## Variance

The results of simulations for estimating  $\sigma^2$  for a budget of  $L = 500$  and  $C_x = .05$  are in Figure 4.3. The results for  $L = 100$  and  $C_x = .1$  are similar and can be found in the Appendix. Similar to the results in the previous sections,  $\lambda^{aux}(L)$  always had a significantly lower MSE compared to  $\lambda^*(L)$ . In all scenarios, the MSE of  $\lambda^{aux}(\widehat{L})$  was also always either significantly less than or not significantly different from that of  $\lambda^*(L)$  while using a smaller budget.

## 4.4 Simulation experiments

In this section, we assess the performance of the auxiliary bootstrap quantile estimator in two simulation experiments. In both experiments, we define the functional of interest,  $F_n$ , the surrogate functional,  $S_n$ , and their respective computational costs,  $c_F$  and  $c_S$ . We simulate the ‘true’  $\alpha \times 100$  quantile of the conditional distribution of  $\widehat{F}_n^*$ , using 10000 bootstrap resamples and computing  $(\widehat{Q}_n^*)^{-1}(\alpha; 10000)$ . We implicitly assume 10000 is close enough to infinity to adequately obtain  $(\widehat{Q}_n^*)^{-1}(\alpha; \infty)$ . In both experiments, our objective is to compare the auxiliary quantile bootstrap against the empirical quantile bootstrap. Because the focus here is on estimating  $(Q)^{-1}(\alpha)$ , we will drop the  $\lambda$  symbol.

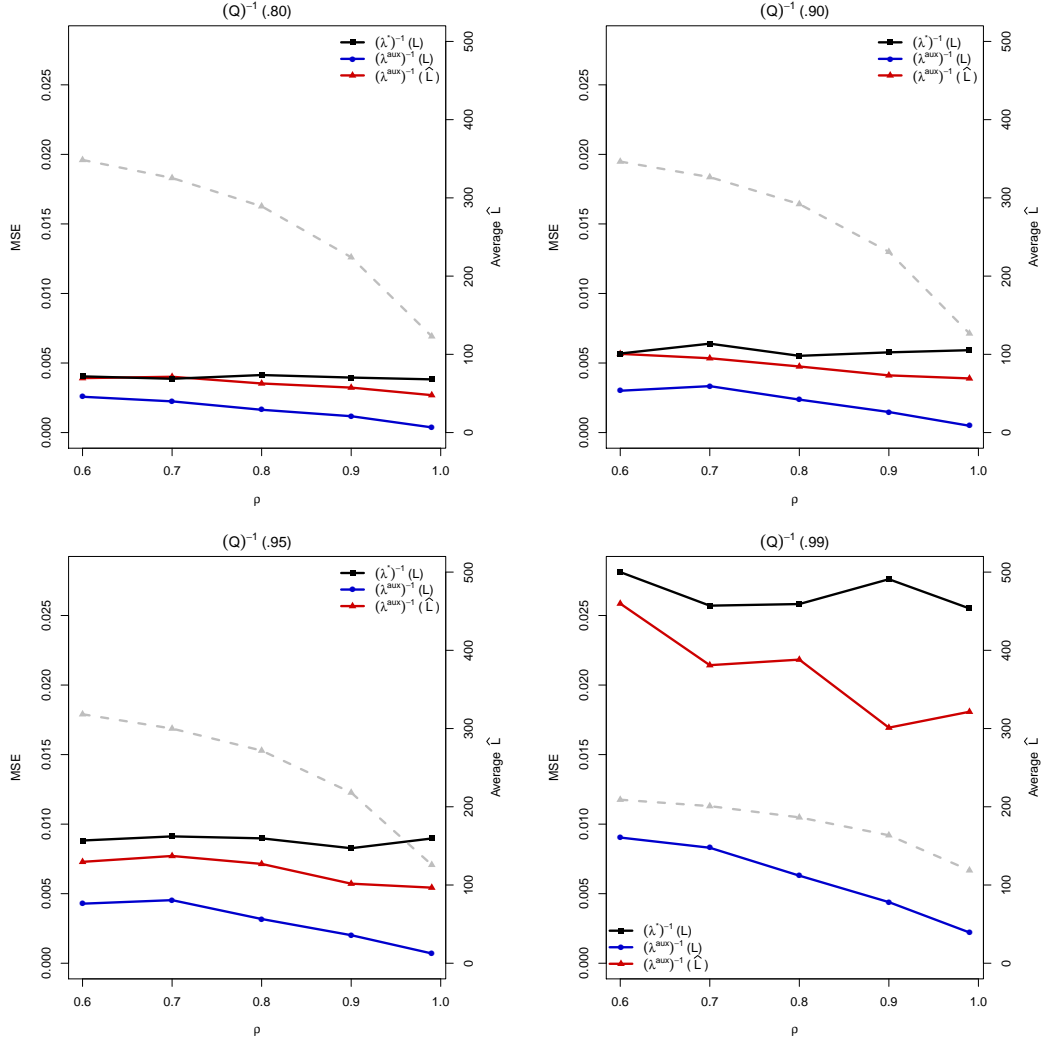


Figure 4.1: Plots of the MSE of  $(\hat{\lambda}^*)^{-1}(L)$ ,  $(\hat{\lambda}^{aux})^{-1}(L)$ , and  $(\hat{\lambda}^{aux})^{-1}(\hat{L})$  for estimating quantiles at  $\alpha \in \{.80, .9, 95, .99\}$  across increasing  $\rho$  where  $L = 500$  and  $C_x = .05$ . Also included is the average computational cost,  $\hat{L}$ , used when computing  $(\hat{\lambda}^{aux})^{-1}(\hat{L})$  (dashed line).

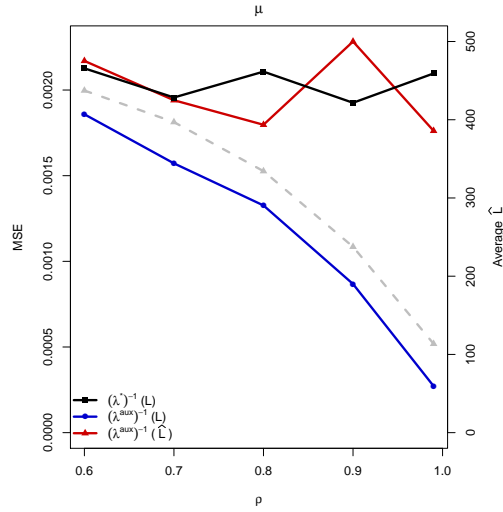


Figure 4.2: Plot of the MSE of  $(\hat{\lambda}^*)^{-1}(L)$ ,  $(\hat{\lambda}^{aux})^{-1}(L)$ , and  $(\hat{\lambda}^{aux})^{-1}(\hat{L})$  for estimating  $\mu$  across increasing  $\rho$  where  $L = 500$  and  $C_x = .05$ . Also included is the average computational cost,  $\hat{L}$ , used when computing  $(\hat{\lambda}^{aux})^{-1}(\hat{L})$  (dashed line).

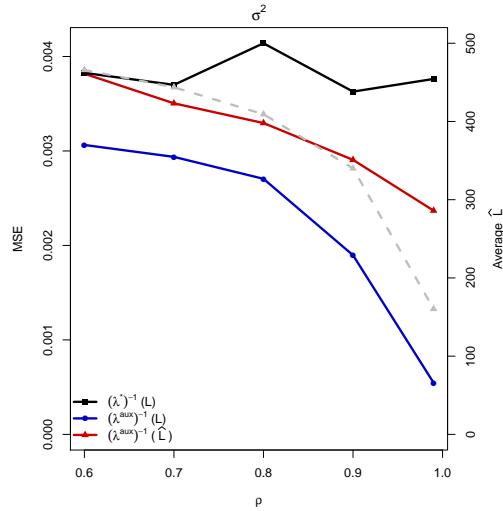


Figure 4.3: Plot of the MSE of  $(\hat{\lambda}^*)^{-1}(L)$ ,  $(\hat{\lambda}^{aux})^{-1}(L)$ , and  $(\hat{\lambda}^{aux})^{-1}(\hat{L})$  for estimating  $\sigma^2$  across increasing  $\rho$  where  $L = 500$  and  $C_x = .05$ . Also included is the average computational cost,  $\hat{L}$ , used when computing  $(\hat{\lambda}^{aux})^{-1}(\hat{L})$  (dashed line).

### 4.4.1 Bayesian regression model

We assume

$$y_i = \mathbf{x}_i^T \boldsymbol{\beta} + \epsilon_i, \quad i = 1, \dots, n, \quad (4.8)$$

where  $X_i = [y_i, \mathbf{x}_i^T]$  are independently and identically distributed random variables,  $\boldsymbol{\beta} = (\beta_0, \beta_1, \beta_2)^T \in \mathbb{R}^3$  are unknown regression coefficients,  $\epsilon_i \sim \text{Normal}(0, \sigma^2)$ ,  $\sigma^2 \in \mathbb{R}^+$ , and  $\mathbf{x}_i = [1, x_{1i}, x_{2i}]$ , where  $x_{1i}$  and  $x_{2i}$  are mutually independent random variables independent of  $\epsilon_i$ . For our simulations, we choose  $n = 100$ ,  $\boldsymbol{\beta} = (10, 10, 10)^T$ ,  $\sigma^2 = 5$ , and we generate  $x_{1i}$  and  $x_{2i}$  from a  $\text{Uniform}(-1, 1)$ .

The parameter of interest is  $\beta_1$ . We define  $F_n$  to be the posterior mode obtained from fitting a Bayesian regression model using either a noninformative or informative prior. We assume *a priori*  $\boldsymbol{\beta} \sim \text{Normal}(\mathbf{0}, \mathbf{B}_0)$ , where  $\mathbf{B}_0$  is

$$\begin{bmatrix} 10 & 0 & 0 \\ 0 & 10 & 0 \\ 0 & 0 & 10 \end{bmatrix} \text{ or } \begin{bmatrix} 10 & 0 & 0 \\ 0 & 1.5 & 0 \\ 0 & 0 & 10 \end{bmatrix},$$

for the noninformative and informative cases respectively. The only difference between the two priors is that the variance of  $\beta_1$  *a priori* is much smaller for the informative prior. For both cases we assume *a priori*  $\sigma^{-2} \sim \text{Gamma}(a_0, b_0)$ , where  $a_0 = .0005$  and  $b_0 = .0005$ , the default diffuse prior parameters provided in the `MCMCregress` function found in the `MCMCpack` in R. Let  $S_n$  be a maximum likelihood counterpart assuming (4.8). The actual time it takes to compute the two functionals will likely differ depending on the software and machine used but it is reasonable to assume that the Bayesian estimator will take significantly longer due to the Markov-chain Monte Carlo algorithm needed to obtain draws from the posterior distribution.

To compute  $F_n$ , we obtain 10000 draws from the posterior distribution of  $\beta_1$  after a burn period of 1000. All computations are to be performed using the `MCMCregress` function. We calculate  $S_n$  using the `glm` function in R. An illustration of the relationship between values of  $F_n$  and  $S_n$  computed on 250 Monte Carlo data sets can be found in Figure 4.4. Clearly, the maximum likelihood based statistic is a better surrogate for the Bayesian estimator when a noninformative prior is used. The average computing time to compute the Bayesian estimator on our machine was .092 seconds (using either the noninformative or informative prior) while the average time to compute the maximum likelihood based estimator was .0016 seconds. Thus the computational cost of  $S_n$  is roughly 57 times cheaper than that of  $F_n$ . In our simulations, we set  $c_F = 1$  and  $c_S = 1/57$ , and thus defined the budget  $L$  in terms of units of  $F_n$ .

In our experiment, we generate  $M_1$  original Monte Carlo data sets,  $\mathbf{X}_1, \dots, \mathbf{X}_{M_1}$ . For each  $\mathbf{X}_i$ , we first obtain  $B_1 = 10000$  bootstrap resamples,  $\mathbf{X}_{ib}^*$ ,  $b = 1, \dots, B_1$ , and compute  $\widehat{F}_n^*$  on each. We then approximate the ‘ideal’ estimator  $\left(\widehat{Q}_n^*\right)^{-1}(\alpha; B_1)^{(i)}$ . Next we compute  $M_2$  conditional Monte Carlo replicates of (1) the empirical quantile  $\left(\widehat{Q}_n^*\right)^{-1}(\alpha, L)^{(ij)}$ , (2) the auxiliary quantile  $\left(\widehat{Q}_n^{aux}\right)^{-1}(\alpha, L)^{(ij)} =$

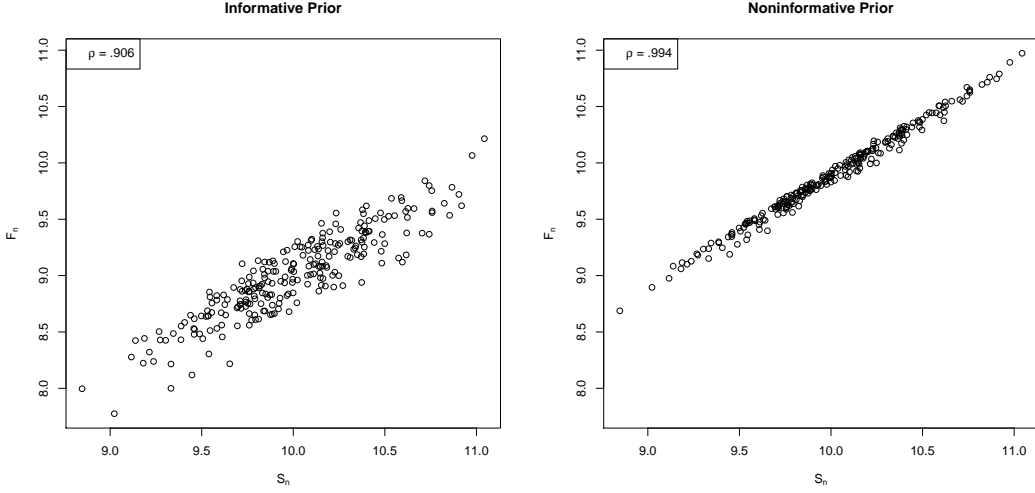


Figure 4.4: Plots of  $F_n$  against  $S_n$  computed on 250 Monte Carlo data sets.

$(\widehat{Q}_n^{aux})^{-1}(\alpha; \widehat{M}_{n, M_0}(L, \alpha), \widehat{K}_{n, M_0}(L, \alpha))^{(ij)}$ , and (3) the potentially less expensive auxiliary quantile

$$(\widehat{Q}_n^{aux})^{-1}(\alpha, \widehat{L})^{(ij)} = (\widehat{Q}_n^{aux})^{-1} \left[ \alpha; \widehat{M}_{n, M_0} \left\{ \widehat{L}_{n, M_0}(\alpha; B), \alpha \right\}, \widehat{K}_{n, M_0} \left\{ \widehat{L}_{n, M_0}(\alpha; B), \alpha \right\} \right]^{(ij)},$$

for  $j = 1, \dots, M_2$ . For the auxiliary estimators, an original sample  $\{F_n^*, S_n^*\}_{i=1}^{25}$  is obtained to determine appropriate values of  $M$  and  $K$ . We measure the conditional performance of each estimator by the conditional mean squared error (MSE), i.e. for the empirical quantile,

$$\text{MSE}^{(i)} \left\{ (\widehat{Q}_n^*)^{-1}(\alpha, L) \right\} = M_2^{-1} \sum_{j=1}^{M_2} \left\{ (\widehat{Q}_n^*)^{-1}(\alpha, L)^{(ij)} - (\widehat{Q}_n^*)^{-1}(\alpha, B_1)^{(i)} \right\}^2;$$

and measure the unconditional performance by the overall average mean squared error (AMSE)

$$\text{AMSE} \left\{ (\widehat{Q}_n^*)^{-1}(\alpha, L) \right\} = M_1^{-1} \sum_{i=1}^{M_1} \text{MSE}^{(i)} \left\{ (\widehat{Q}_n^*)^{-1}(\alpha, L) \right\}.$$

For our simulations, we fix  $M_1 = 100$  and  $M_2 = 100$ , thus generating a total of 10000 Monte Carlo replicates of each estimator. We consider  $\alpha \in \{.8, .95\}$  and  $L \in \{50, 100, 250\}$ .

The results of the simulation experiment are found in Figure 4.5. For all values of  $\alpha$  and each computational budget,  $(\widehat{Q}_n^{aux})^{-1}(\alpha, L)$  had a significantly ( $\alpha = .01$ ) lower AMSE than  $(\widehat{Q}_n^*)^{-1}(\alpha, L)$ . This was particularly true when  $F_n$  was computed using the noninformative prior as  $S_n$  was a better surrogate in that scenario. The plots also show that the benefits of  $(\widehat{Q}_n^{aux})^{-1}(\alpha, L)$  are generally not as strong as  $L$  increases. The AMSE of  $(\widehat{Q}_n^{aux})^{-1}(\alpha, \widehat{L})$  was always either significantly less than or not

significantly different from that of  $(\widehat{Q}_n^*)^{-1}(\alpha, L)$ . The reduced cost is represented by the average value of  $\widehat{L}$  found on the dashed line. The auxiliary estimator was able to achieve the same (or better) level of precision as that of the standard empirical quantile estimator, at a fraction of the cost. This was particularly true in the noninformative case.

#### 4.4.2 Geospatial weather model

Let  $Y(\mathbf{s})$  denote a dependent variable on a two-dimensional spatial grid  $\mathbf{s} \in D$  for some region  $D \subset \mathbb{R}^2$ . We assume  $Y(\mathbf{s}) = \mu + \omega(\mathbf{s}) + \epsilon(\mathbf{s})$ , where  $\mu$  is the overall mean,  $\epsilon(\mathbf{s}) \sim N(0, \tau^2)$  is the error, independent of  $Y(\mathbf{s})$ , and  $\omega(\mathbf{s})$  is the spatial error which accounts for a possible spatial dependence. The error  $\omega(\mathbf{s})$  is typically assumed to be a Gaussian process governed by a small selection of unknown parameters, denoted by  $\gamma$ . Some possible covariance models for  $\text{Cov}\{\omega(\mathbf{s}), \omega(\mathbf{s}')\}$  are the Matern, Cauchy, and exponential. Let  $\mathbf{s}$  be composed of  $n$  locations,  $\mathbf{s}^{(1)}, \mathbf{s}^{(2)}, \dots, \mathbf{s}^{(n)}$ , and thus  $\mathbf{Y}(\mathbf{s}) = \{Y(\mathbf{s}^{(1)}), Y(\mathbf{s}^{(2)}), \dots, Y(\mathbf{s}^{(n)})\}'$  where  $\mathbf{Y}(\mathbf{s}) \sim N\{\mathbf{1}_n \mu, \Sigma(\gamma)\}$ , for some covariance matrix  $\Sigma(\gamma) = \mathbf{C} + \tau^2 \mathbf{I}_n$ , where  $C(i, j) = \text{Cov}\{\omega(\mathbf{s}^{(i)}), \omega(\mathbf{s}^{(j)})\}$ . The resulting log-likelihood without scalars that do not depend on  $\mu$  or  $\gamma$  is  $l(\mathbf{Y}(\mathbf{s}); \mu, \gamma) = -\frac{1}{2} \log|\Sigma(\gamma)| - \frac{1}{2} \{\mathbf{Y}(\mathbf{s}) - \mathbf{1}_n \mu\}' \Sigma(\gamma)^{-1} \{\mathbf{Y}(\mathbf{s}) - \mathbf{1}_n \mu\}$ . As  $n$  gets large, this log-likelihood gets increasingly difficult to evaluate as it requires taking the determinant and inverse of  $\Sigma(\gamma)$ . Thus, obtaining an estimate of  $\mu$  or  $\gamma$  is computationally expensive for large  $n$ .

[18] propose a less expensive alternative to evaluating the full log-likelihood based on a block composite likelihood. They partition  $D$  into  $d$  spatial blocks where the log-likelihood is evaluated in pairs of blocks which are then summed to approximate the overall log-likelihood. The idea is to split up the likelihood for faster computation while preserving most of the spatial dependencies in the data. As more blocks are used, the computational complexity decreases while the overall efficiency of estimating  $\mu$  or  $\gamma$  also decreases. For a more thorough discussion on these methods, see [18].

Let  $\mu$  be the parameter of interest and  $F_n = \widehat{\mu}$  be the estimate of  $\mu$  based on fitting the geostatistical model using the full likelihood. We define  $S_n^d$  to be the estimate of  $\mu$  using the block composite likelihood method where  $d$  blocks are used. Thus  $S_n^1 = F_n$ .

We generate  $\mathbf{s}$  by randomly sampling  $n = 1000$  points from a uniform  $(0, 1) \times (0, 1)$  spatial domain. We assume an exponential covariance model for  $\Sigma(\gamma)$  of the form  $\text{Cov}\{\omega(\mathbf{s}^{(i)}), \omega(\mathbf{s}^{(j)})\} = \tau^2 I(h = 0) + \sigma^2 \exp(-1/\rho \times h)$  where  $h = \|\mathbf{s}^{(i)} - \mathbf{s}^{(j)}\|$ . Thus  $\gamma = \{\tau^2, \sigma^2, \rho\}$ . For our simulations, we fix  $\mu = 0$  and  $\gamma = \{1, 1, .03\}$ . Figure 4.6 displays, for increasing  $d$ , both the sample correlation between  $F_n$  and  $S_n^d$  as well as the average computation time (in seconds) to compute  $S_n^d$  based on 100 data sets. We consider blocks of size  $d = 1, 2^2 = 4, 3^2 = 9, 4^2 = 16, \dots, 10^2 = 100$ . The correlation is highest at  $d = 9^2 = 81$ , thus we define  $S_n = S_n^{81}$ .

In our simulations, we generate  $M_1$  original Monte Carlo data sets,  $\mathbf{Y}_1(\mathbf{s}_1), \dots, \mathbf{Y}_{M_1}(\mathbf{s}_{M_1})$ . For each  $\mathbf{Y}_i(\mathbf{s}_i)$ , we first obtain  $B_1 = 10000$  bootstrap resamples,  $\mathbf{Y}_{ib}^*(\mathbf{s}_i)$ ,  $b = 1, \dots, B_1$ , and compute  $\widehat{F}_n^*$  on each. We then approximate the ‘ideal’ estimator  $(\widehat{Q}_n^*)^{-1}(\alpha; B_1)^{(i)}$  for  $i = 1, \dots, M_1$ . Because the data are spatially correlated, these bootstrap samples are parametric bootstrap resamples. We use estimates of  $\mu$  and  $\gamma$  from  $\mathbf{Y}_m(\mathbf{s}_m)$  to obtain a random sample using the generative model described previously where the true values of  $\mu$  and  $\gamma$  are replaced by their corresponding estimates. Next we compute the

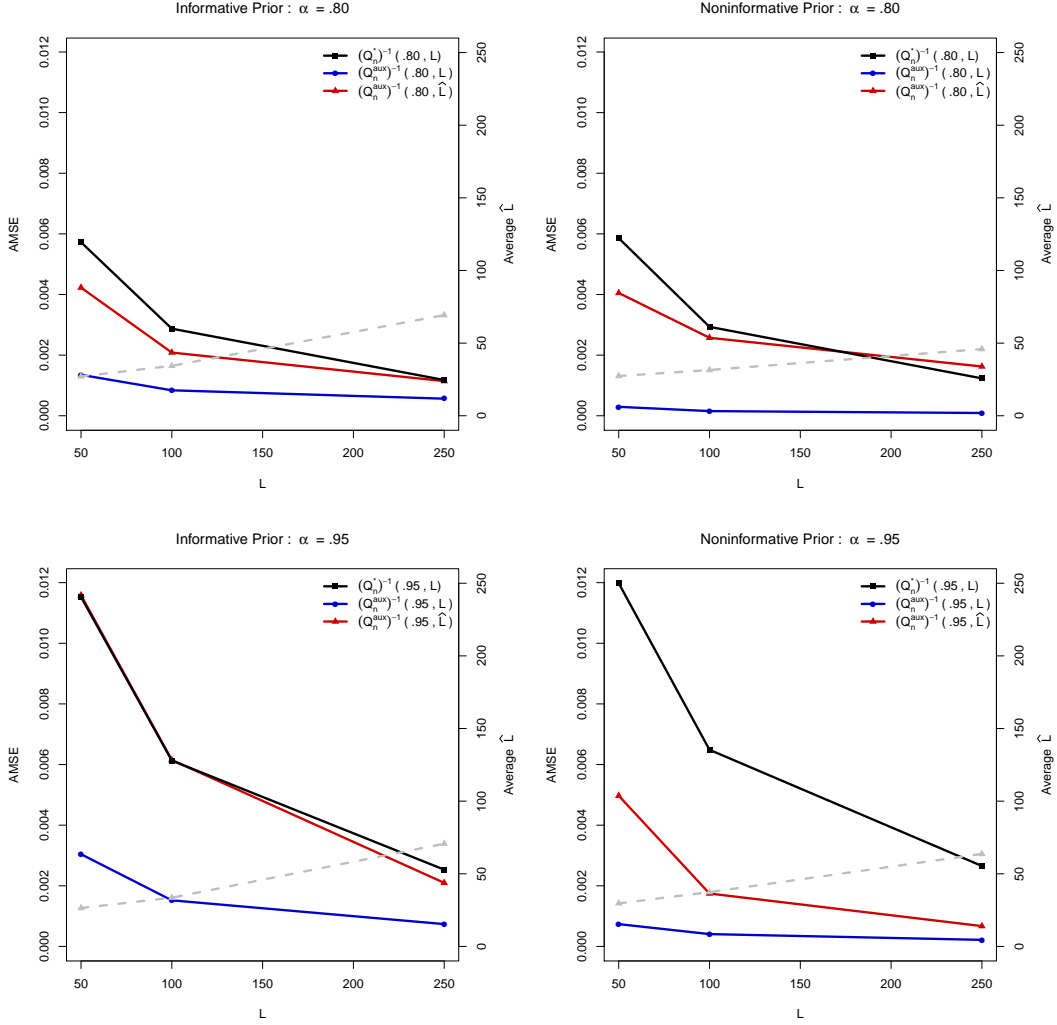


Figure 4.5: Plots of the AMSE of  $(\hat{Q}_n^*)^{-1}(\alpha, L)$ ,  $(\hat{Q}_n^{aux})^{-1}(\alpha, L)$ , and  $(\hat{Q}_n^{aux})^{-1}(\alpha, \hat{L})$  for  $\alpha \in \{.80, .95\}$  and increasing computational budgets. Also included is the average computational cost,  $\hat{L}$ , used when computing  $(\hat{Q}_n^{aux})^{-1}(\alpha, \hat{L})$  (dashed line). The plots on the left display results for when  $F_n$  is computed assuming an informative prior while the plots on the right display results for the noninformative prior.

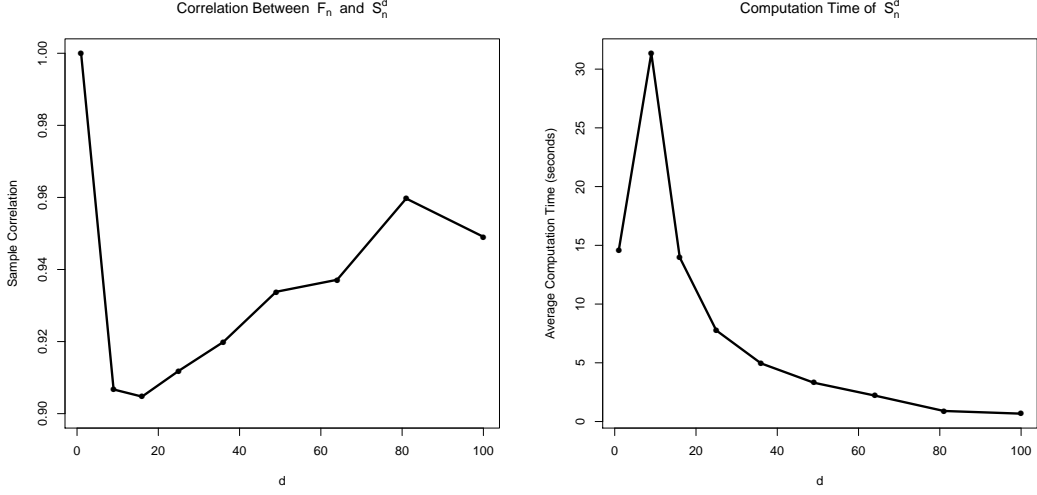


Figure 4.6: Plots which show the correlation between  $F_n$  and  $S_n$  (left) and the average computation time of  $S_n^d$  (right) for increasing values of  $d$  based on 100 Monte Carlo data sets.

empirical quantile  $(\widehat{Q}_n^*)^{-1}(\alpha, L)^{(ij)}$ , and the auxiliary quantile estimators

$$(\widehat{Q}_n^{aux})^{-1}(\alpha, L)^{(ij)} = (\widehat{Q}_n^{aux})_d^{-1}(\alpha; \widehat{M}_{n, M_0}(L, \alpha), \widehat{K}_{n, M_0}(L, \alpha))^{(ij)}$$

and

$$(\widehat{Q}_n^{aux})^{-1}(\alpha, \widehat{L})^{(ij)} = (\widehat{Q}_n^{aux})_d^{-1}[\alpha; \widehat{M}_{n, M_0} \{ \widehat{L}_{n, M_0}(\alpha; B), \alpha \}, \widehat{K}_{n, M_0} \{ \widehat{L}_{n, M_0}(\alpha; B), \alpha \}]^{(ij)},$$

for  $j = 1, \dots, M_2$ . We fix  $M_1 = 20$  and  $M_2 = 50$ , thus generating a total of 1000 Monte Carlo replicates of each estimator. We consider  $\alpha \in \{.8, .95\}$  and  $L \in \{50, 100, 250\}$ .

The results of the simulation experiment are found in Figure 4.7. The AMSE of  $(\widehat{Q}_n^{aux})^{-1}(\alpha, L)$  was always significantly less than ( $\alpha = .01$ ) than that of  $(\widehat{Q}_n^*)^{-1}(\alpha, L)$ , particularly when estimating the 95th quantile. The AMSE of  $(\widehat{Q}_n^{aux})^{-1}(\alpha, \widehat{L})$  was never significantly different ( $\alpha = .01$ ) than  $(\widehat{Q}_n^*)^{-1}(\alpha, L)$  yet was able to achieve the same level of precision at a reduced cost (over layed dashed line).

## 4.5 Discussion

The bootstrap is a simple, yet powerful, method that uses Monte Carlo sampling to approximate the quantile, mean, or variance of the sampling distribution of a functional of interest. Its usefulness is sometimes limited if the functional is computationally expensive. In this section, we introduced a method that incorporates a correlated, less expensive, functional which allows us to either 1) estimate the quantile,

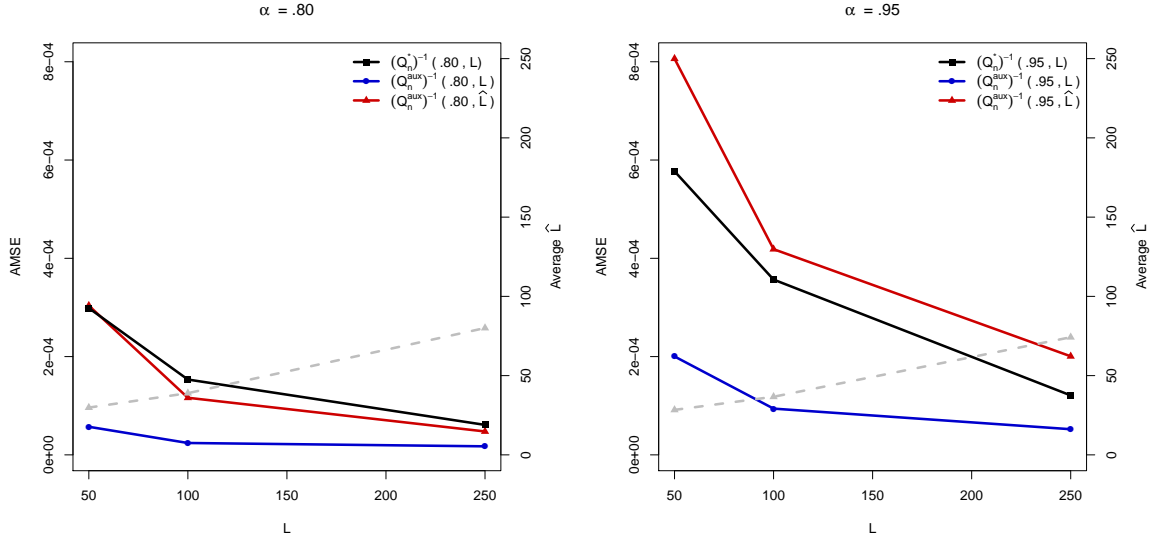


Figure 4.7: Plots of the AMSE of  $(\hat{Q}_n^*)^{-1}(\alpha, L)$ ,  $(\hat{Q}_n^{aux})^{-1}(\alpha, L)$ , and  $(\hat{Q}_n^{aux})^{-1}(\alpha, \hat{L})$  for  $\alpha \in \{.80, .95\}$  and increasing computational budgets. Also included is the average computational cost,  $\hat{L}$ , used when computing  $(\hat{Q}_n^{aux})^{-1}(\alpha, \hat{L})$  (dashed line).

mean, or variance, more precisely assuming a fixed computational budget, or 2) estimate the same quantity with the same precision as a standard empirical estimator at a fraction of the cost. We performed a proof of concept simulation study which showed the benefits of the auxiliary approach increased as the correlation between  $F_n$  and  $S_n$  approached 1. We then conducted two simulation experiments comparing the auxiliary quantile bootstrap estimators for increasing computational budgets. In all cases, under a fixed computational budget, the auxiliary bootstrap had a significantly lower AMSE than the standard quantile estimator, particularly so if the computational budget is small. Our simulations also showed that in almost all cases, the less-expensive auxiliary bootstrap estimator was able to achieve the same level of precision, or better, as the standard estimator, but at a reduced cost.

## Chapter 5

# Conclusion

The bootstrap has been a fundamental tool used by researchers and statisticians to approximate attributes of the sampling distribution of a functional of interest. It allows a user to bypass complex derivations and instead rely on repeated Monte Carlo sampling. In the age of big data, the complexity of estimators has increased, making Monte Carlo sampling computationally burdensome. In this work, we introduced a method to alleviate this burden. We proposed leveraging a surrogate functional that is correlated with the functional of interest yet computationally less expensive in what we called ‘auxiliary bootstrap’ estimators.

Unlike the auxiliary mean and variance estimators constructed from the laws of total expectation and variance, the auxiliary quantile estimator was not readily apparent. We thus borrowed from the sampling literature and conducted an extensive literature review of quantile estimators that leverage auxiliary information. This review is found in Chapter 2. We found that there was a vast amount of estimators and thus conducted a simulation experiment to compare them all. We found that generally, the Chambers and Dunstan model-based estimator was the preferred estimator, especially if the relationship between the functional of interest and the surrogate was well captured by a statistical model.

In addition to the literature review and simulation experiment, we created an R package `auxQuantile` which allows a user to compute all of the auxiliary quantile estimators described for their data. A description of this package is in Chapter 3.

In Chapter 4, we formally introduced the auxiliary bootstrap. We first performed a proof of concept simulation study that showed the benefit of the auxiliary quantile, mean, and variance estimators for two correlated random variables. We hypothetically made one of the variables to be more ‘expensive’, emulating the functional of interest, and the other variable less ‘expensive’, emulating the surrogate functional. This allowed us to explore the auxiliary approach for varying levels of the correlation between the two variables, as well as varying computational costs and budgets. The results showed that as the correlation between the two variables increased, the auxiliary estimators were more precise and in some cases computationally cheaper.

Next, we performed two simulation experiments with the focus on the auxiliary quantile bootstrap. The first experiment involved a linear Bayesian regression model where the functional of interest was an estimated slope coefficient and the correlated surrogate was a computationally cheaper frequen-

tist maximum likelihood-based counterpart. The simulation results showed that the auxiliary bootstrap was always significantly more precise than the standard bootstrap under a fixed computational budget. This showed that the auxiliary bootstrap was able to use resources more effectively and more efficiently. The results also showed that the auxiliary quantile was able to achieve a desired precision for a fraction of the cost as the standard empirical quantile estimator.

The final simulation experiment involved a geospatial weather model where the functional of interest was an estimate of a mean parameter from a fitted spatial model. The surrogate functional was a similar estimator of the mean based on a computationally cheaper block-composite likelihood method. The simulation results again showed that the auxiliary bootstrap always outperformed the standard approach under a fixed budget. Additionally, in almost all cases, the auxiliary bootstrap was able to achieve a pre-specified level of precision at a reduced computational cost.

## REFERENCES

- [1] Amparo Albalade and Wolfgang Minker. *Semi-Supervised and Unsupervised Learning: Novel Strategies*. John Wiley and Sons, Inc., Hoboken, NJ, USA, 2013.
- [2] Donald W. K. Andrews and Moshe Buchinsky. A three-step method for choosing the number of bootstrap repetitions. *Econometrica*, 68(1):23–51, January 2000.
- [3] Dimitri P Bertsekas. *Nonlinear programming*. Athena scientific Belmont, 1999.
- [4] David Bollier and Charles M. Firestone. The promise and peril of big data. Washington, DC: Aspen Institute, Communications and Society Program, 2010.
- [5] Emanuele Borgonovo, William Castaings, and Stefano Tarantola. Model emulation and moment-independent sensitivity analysis: An application to environmental modelling. *Environmental Modelling & Software*, 34:105–115, 2012.
- [6] Jeffrey S Buzas. Fast estimators of the jackknife. *The American Statistician*, 51(3):235–240, 1997.
- [7] Bradley P Carlin and Thomas A Louis. *Bayes and empirical Bayes methods for data analysis*, volume 17. Chapman & Hall/CRC Boca Raton, FL, 2000.
- [8] R. L. Chambers and R. Dunstan. Estimating distribution functions from survey data. *Biometrika*, 73(3):597–604, 1986.
- [9] Raymond L. Chambers, Alan H. Dorfman, and Thomas E. Wehrly. Bias robust estimation in finite populations using nonparametric calibration. *Journal of the American Statistical Association*, 88(421):268–277, 1993.
- [10] Olivier Chapelle, Bernhard Scholkopf, and Alexander Zien. *Semi-Supervised Learning*. The MIT Press, Cambridge MA, USA, 2006.
- [11] Jiahua Chen and Changbao Wu. Estimation of distribution function and quantiles using the model-calibrated pseudo empirical likelihood method. *Statistica Sinica*, 12(4):1223–1239, 2002.
- [12] William G. Cochran. *Sampling Techniques, Third Edition*. John Wiley and Sons, Inc., Hoboken, NJ, USA, 1977.
- [13] Jean-Claude Deville and Carl-Erik Sarndal. Calibration estimators in survey sampling. *Journal of the American Statistical Association*, 87(418):376–382, 1992.

- [14] Alan H. Dorfman. A comparison of design-based and model-based estimators of the finite population distribution function. *Australian and New Zealand Journal of Statistics*, 35:29–41, 1993.
- [15] Alan H. Dorfman and Peter Hall. Estimators of the finite population distribution function using nonparametric regression. *The Annals of Statistics*, 21(3):1452–1475, 1993.
- [16] M. Duggan and J Brenner. Pew research center’s internet and american life project. Pew Research Center, Washington, DC, USA., 2013.
- [17] Bradley Efron and Trevor Hastie. *Computer Age Statistical Inference*, volume 5. Cambridge University Press, 2016.
- [18] Jo Eidsvik, Benjamin A. Shaby, Brian J. Reich, Matthew Wheeler, and Jarad Niemi. Estimation and prediction in spatial models with block composite likelihoods. *Journal of Computational and Graphical Statistics*, 23(2):295–315, 2013.
- [19] Bradley Ferguson, Eric Laber, and Leonard Stefanski. *auxQuantile: Auxiliary Quantile Estimation*. R Package Version 1.1, 2016.
- [20] Bradley Ferguson, Eric Laber, and Leonard Stefanski. Quantile estimation in the presence of auxiliary information: An overview. 2016.
- [21] Carol A. Francisco and Wayne A. Fuller. Quantile estimation with a complex survey design. *Annals of Statistics*, 19:454–469, 1991.
- [22] Wayne A. Fuller. *Sampling Statistics*. John Wiley and Sons, Inc., Hoboken, NJ, USA, 2009.
- [23] V. P. Godambe. Estimation of cumulative distribution of a survey population. Technical report, Department of Statistics and Actuarial Science, University of Waterloo., 1989.
- [24] Radek Grzeszczuk, Demetri Terzopoulos, and Geoffrey Hinton. Neuroanimator: Fast neural network emulation and control of physics-based models. In *Proceedings of the 25th annual conference on Computer graphics and interactive techniques*, pages 9–20. ACM, 1998.
- [25] Tim Harford. Big data: A big mistake? *Significance*, 11(5):14–19, 2014.
- [26] Eszter Hargittai. Is bigger always better? potential biases of big data derived from social network sites. *The Annals of the American Academy of Political and Social Science*, 659(1):63–76, 2015.
- [27] Torsten Harms and Pierre Duchesne. On calibration estimation for quantiles. *Survey Methodology*, 32(1):37–52, 2006.

- [28] Miguel Helft. Google uses web searches to track flu's spread., November 2008.
- [29] D. G. Horvitz and D. J. Thompson. A generalization of sampling without replacement from a finite universe. *Journal of the American Statistical Association*, 47(260):663–685, 1952.
- [30] Alicia A. Johnson, F. Jay Breidt, and Jean D. Opsomer. Estimating distribution functions from survey data using nonparametric regression. *Journal of Statistical Theory and Practice*, 2(3):419–431, 2008.
- [31] Leslie Kish. *Survey Sampling*. John Wiley and Sons, Inc., Hoboken, NJ, USA, 1965.
- [32] Rob Kitchin. *The Data Revolution: Big Data, Open Data, Data Infrastructures and Their Consequences*. Sage Publications, 2014.
- [33] Anthony Y. C. Kuk. A kernel method for estimating finite population distribution functions using auxiliary information. *Biometrika*, 80(2):385–392, 1993.
- [34] Anthony Y. C. Kuk and T. K. Mak. Median estimation in the presence of auxiliary information. *Journal of the Royal Statistical Society. Series B (Methodological)*, 51(2):261–269, 1989.
- [35] Lynn Kuo. Classical and prediction approaches to estimating distribution functions from survey data. In *Proceedings of the Section on Survey Research Methods, American Statistical Association*, pages 280–285, 1988.
- [36] David Lazer, Ryan Kennedy, Gary King, and Alessandro Vespignani. The parable of google flu: Traps in big data analysis. *Science*, 343:1203–1205, 2014.
- [37] Erich L. Lehmann. *Fisher, Neyman, and the Creation of Classical Statistics*. Springer, New York, NY, 2011.
- [38] Sharon L. Lohr. *Sampling: Design and Analysis, 2nd Edition*. Brooks/Cole, Boston, MA, 2010.
- [39] S.L. Lohr and T.E. Raghunathan. Combining survey data with other data sources. *Statistical Science*, Forthcoming:1–28, 2017.
- [40] Maria-Jose Lombardia and Wencesiao Gonzalez-Manteiga. Estimation of a finite population distribution function based on a linear model with unknown heteroscedastic errors. *The Canadian Journal of Statistics*, 33(2):181–200, 2005.
- [41] S. Martinez, M. Rueda, A. Arcos, and H. Martinez. Optimum calibration points estimating distribution functions. *Journal of Computational and Applied Mathematics*, 233:2265–2277, 2010.

- [42] S. Martinez, M. Rueda, A. Arcos, H. Martinez, and I. Sanchez-Borrego. Post-stratified calibration method for estimating quantiles. *Computational Statistics and Data Analysis*, 55:838–851, 2011.
- [43] Zbigniew Michalewicz and David B Fogel. *How to solve it: modern heuristics*. Springer Science & Business Media, 2013.
- [44] Leo Pasquazzi and Lucio De Capitani. A new estimator for a finite population distribution function in the presence of complete auxiliary information. 2014.
- [45] Leo Pasquazzi and Lucio de Capitani. A comparison between nonparametric estimators for finite population distribution functions. *Survey Methodology*, 42(1):87–120, 2016.
- [46] J. N. K. Rao. Estimating totals and distribution functions using auxiliary information at the estimation stage. *Journal of Official Statistics*, 10(2):153–165, 1994.
- [47] J. N. K. Rao, J. G. Kovar, and H. J. Mantel. On estimating distribution functions and quantiles from survey data using auxiliary information. *Biometrika*, 77(2):365–375, 1990.
- [48] Marco Ratto, Andrea Castelletti, and Andrea Pagano. Emulation techniques for the reduction and sensitivity analysis of complex environmental models, 2012.
- [49] Tim Robertson, F. T. Wright, and R. L. Dykstra. *Order Restricted Statistical Inference*. Wiley, New York, NY, 1988.
- [50] C. Rudin, D. Dunson, R. Irizarry, H. Ji, E. Laber, J. Jeek, and T. McCormick. Discovery with data: Leveraging statistics with computer science to transform science and society. A Working Group of the American Statistical Association., 2014.
- [51] M. Rueda, S. Martinez, H. Martinez, and A. Arcos. Estimation of the distribution function with calibration methods. *Computational Statistics and Data Analysis.*, 137:435–448, 2007.
- [52] M. Rueda and J. F. Munoz. New model-assisted estimators for the distribution function using the pseudo empirical likelihood method. *Statistica Neerlandica.*, 63(2):227–244, 2009.
- [53] M. Rueda, JI. Sanchez-Borrego, A. Arcos, and S. Martinez. Model-calibration estimation of the distribution function using nonparametric regression. *Metrika.*, 71:33–44, 2010.
- [54] M. Del Mar Rueda, Antonio Arcos, and M. Dolores Martinez. Difference estimators of quantiles in finite populations. *Sociedad de Estadística e Investigación Operativa.*, 12(2):481–496, 2003.

- [55] M. M. Rueda, A. Arcos, M. D. Martinez-Miranda, and Y. Roman. Some improved estimators of finite population quantile using auxiliary information in sample surveys. *Computational Statistics and Data Analysis.*, 45:825–848, 2004.
- [56] David W Scott. *Multivariate density estimation: theory, practice, and visualization*. John Wiley & Sons, 2015.
- [57] David W. Scott. *Nonparametric Regression and Additive Models, in Multivariate Density Estimation: Theory, Practice, and Visualization, Second Edition*. John Wiley and Sons, Inc, Hoboken, NJ, 2015.
- [58] P. L. D. Nascimento Silva and C. J. Skinner. Estimating distribution functions with auxiliary information using poststratification. *Journal of Official Statistics.*, 11(3):277–294, 1995.
- [59] Bernard W. Silverman. *Density Estimation for Statistics and Data Analysis*. Chapman and Hall, London, 1985.
- [60] Housila P. Singh, Sarjinder Singh, and Marcin Kozak. A family of estimators of finite-population distribution function using auxiliary information. *JActa Applicandae Mathematicae.*, 104(2):115–130, 2008.
- [61] R Core Team. *R: A Language and Environment for Statistical Computing*.
- [62] Suojin Wang and Alan H. Dorfman. A new estimator for the finite population distribution function. *Biometrika*, 83(3):639–652, 1996.
- [63] Changbao Wu and Randy R. Sitter. A model-calibration approach to using complete auxiliary information from survey data. *Journal of the American Statistical Association*, 96(453):185–193, 2001.
- [64] Ziaojin Zhu and Andrew B. Goldberg. *Introduction to Semi-Supervised Learning*. Morgan and Claypool Publishers, 2009.

## APPENDIX

# Appendix A

## Additional Results

### A.1 Additional results from the proof of concept simulation experiment

The following plots contain further results for  $L = 100$  and  $C_x = .1$ .

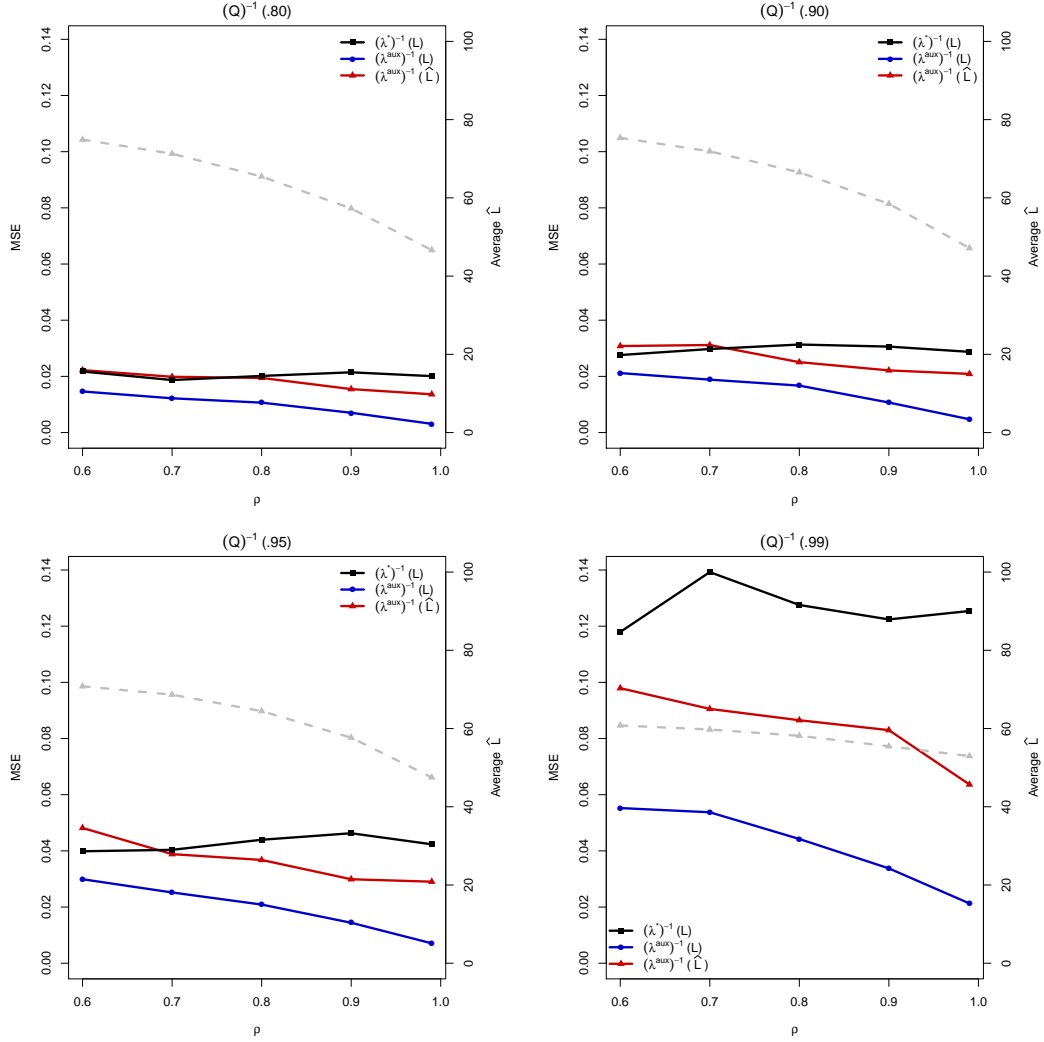


Figure A.1: Plots of the MSE of  $(\hat{\lambda}^*)^{-1}(L)$ ,  $(\hat{\lambda}^{aux})^{-1}(L)$ , and  $(\hat{\lambda}^{aux})^{-1}(\hat{L})$  for estimating quantiles at  $\alpha \in \{.80, .9, .95, .99\}$  across increasing  $\rho$  where  $L = 100$  and  $C_x = .1$ . Also included is the average computational cost,  $\hat{L}$ , used when computing  $(\hat{\lambda}^{aux})^{-1}(\hat{L})$  (dashed line).

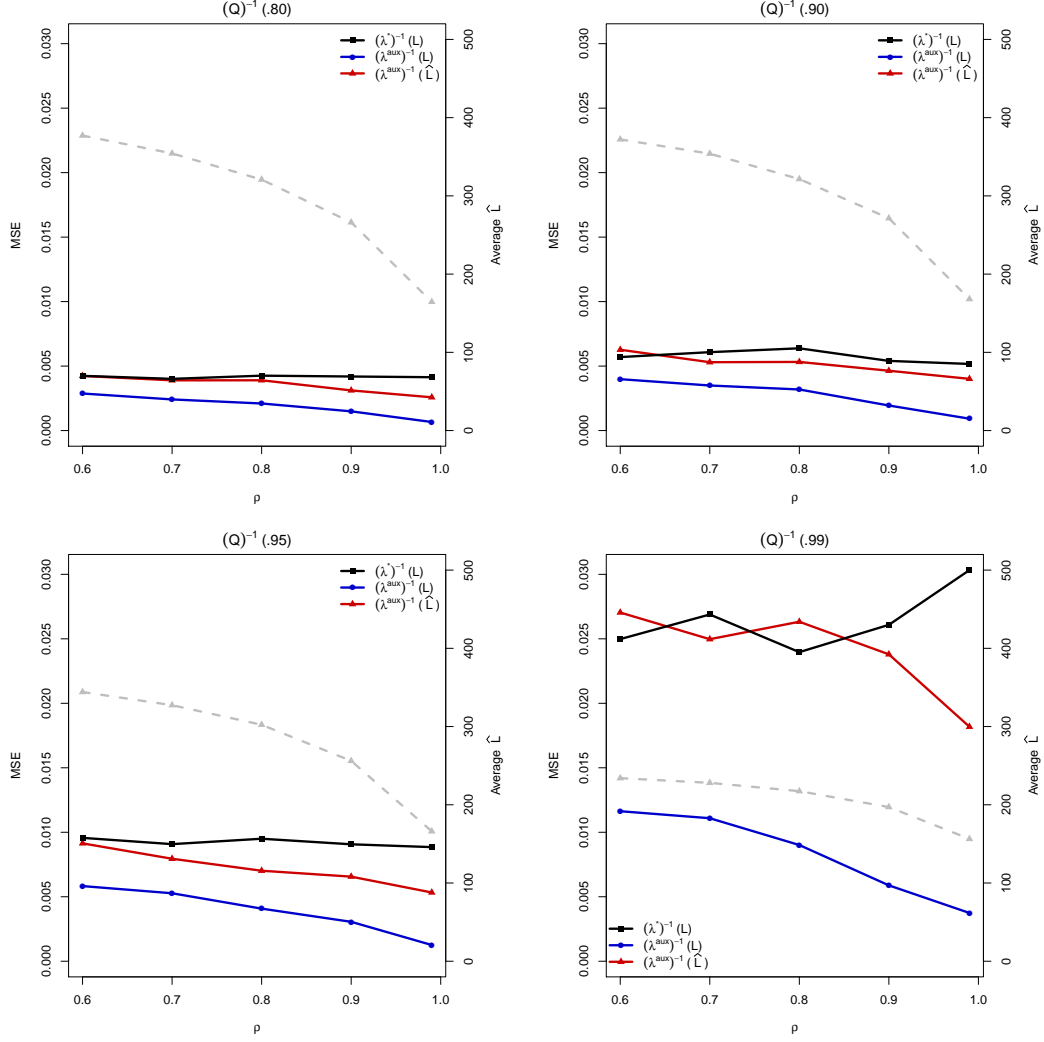


Figure A.2: Plots of the MSE of  $(\hat{\lambda}^*)^{-1}(L)$ ,  $(\hat{\lambda}^{aux})^{-1}(L)$ , and  $(\hat{\lambda}^{aux})^{-1}(\hat{L})$  for estimating quantiles at  $\alpha \in \{.80, .9, 95, .99\}$  across increasing  $\rho$  where  $L = 500$  and  $C_x = .1$ . Also included is the average computational cost,  $\hat{L}$ , used when computing  $(\hat{\lambda}^{aux})^{-1}(\hat{L})$  (dashed line).

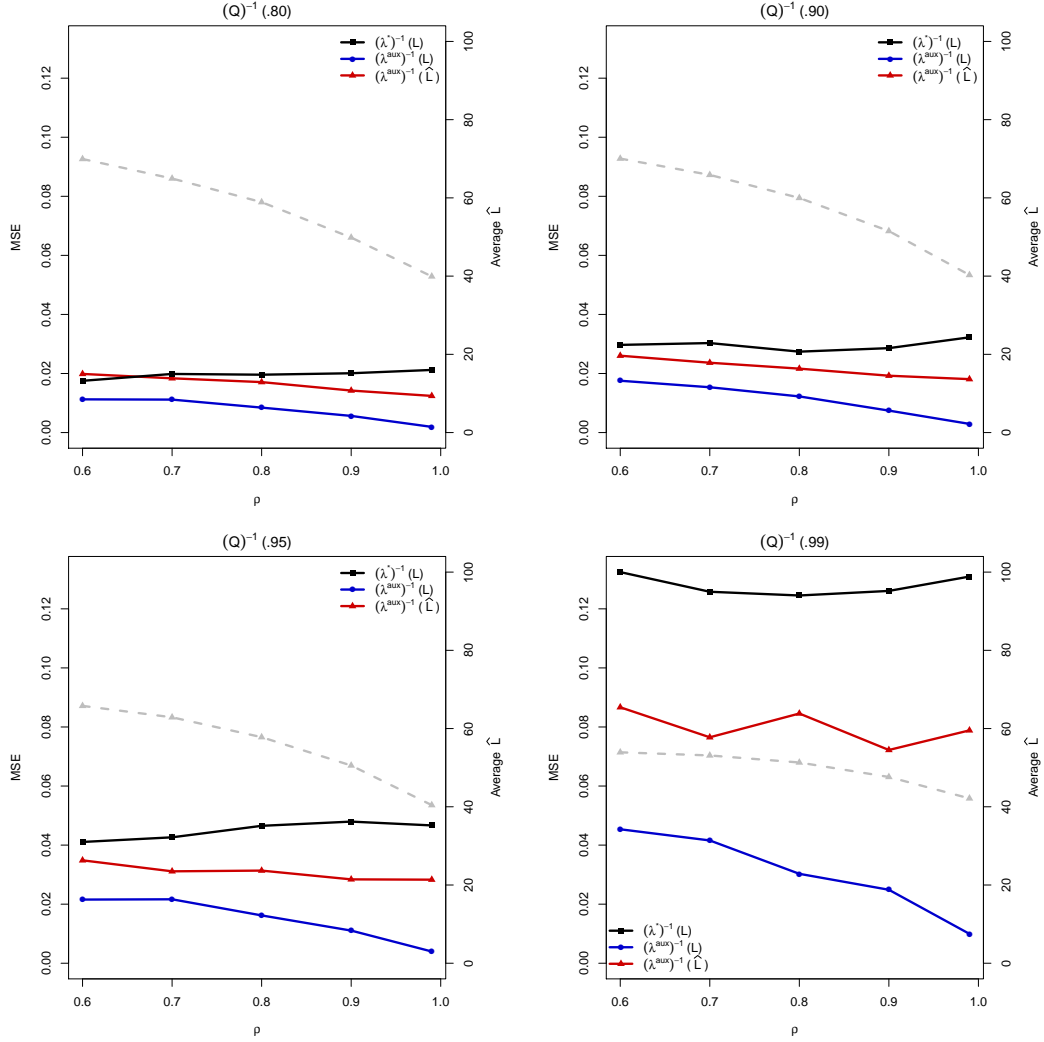


Figure A.3: Plots of the MSE of  $(\hat{\lambda}^*)^{-1}(L)$ ,  $(\hat{\lambda}^{aux})^{-1}(L)$ , and  $(\hat{\lambda}^{aux})^{-1}(\hat{L})$  for estimating quantiles at  $\alpha \in \{.80, .9, .95, .99\}$  across increasing  $\rho$  where  $L = 100$  and  $C_x = .05$ . Also included is the average computational cost,  $\hat{L}$ , used when computing  $(\hat{\lambda}^{aux})^{-1}(\hat{L})$  (dashed line).

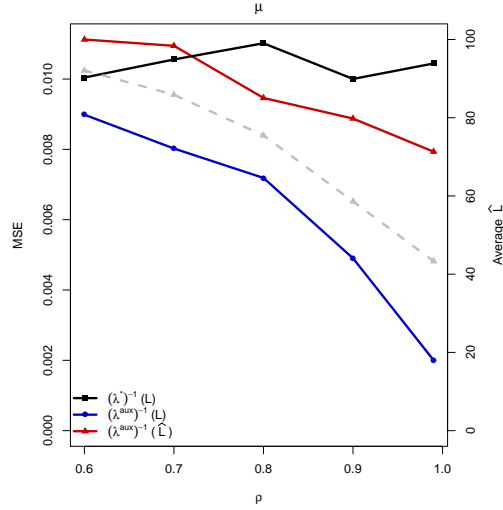


Figure A.4: Plot of the MSE of  $(\hat{\lambda}^*)^{-1}(L)$ ,  $(\hat{\lambda}^{aux})^{-1}(L)$ , and  $(\hat{\lambda}^{aux})^{-1}(\hat{L})$  for estimating  $\mu$  across increasing  $\rho$  where  $L = 100$  and  $C_x = .1$ . Also included is the average computational cost,  $\hat{L}$ , used when computing  $(\hat{\lambda}^{aux})^{-1}(\hat{L})$  (dashed line).

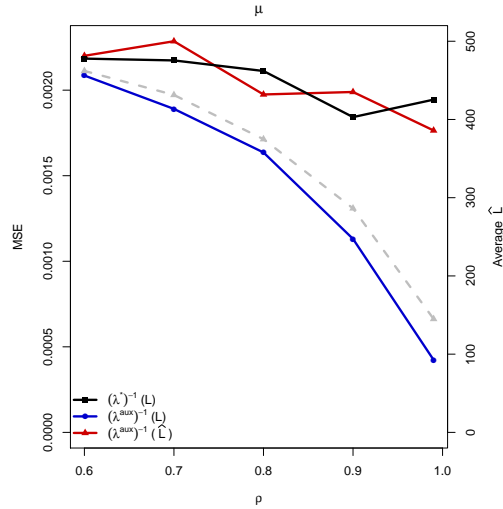


Figure A.5: Plot of the MSE of  $(\hat{\lambda}^*)^{-1}(L)$ ,  $(\hat{\lambda}^{aux})^{-1}(L)$ , and  $(\hat{\lambda}^{aux})^{-1}(\hat{L})$  for estimating  $\mu$  across increasing  $\rho$  where  $L = 500$  and  $C_x = .1$ . Also included is the average computational cost,  $\hat{L}$ , used when computing  $(\hat{\lambda}^{aux})^{-1}(\hat{L})$  (dashed line).

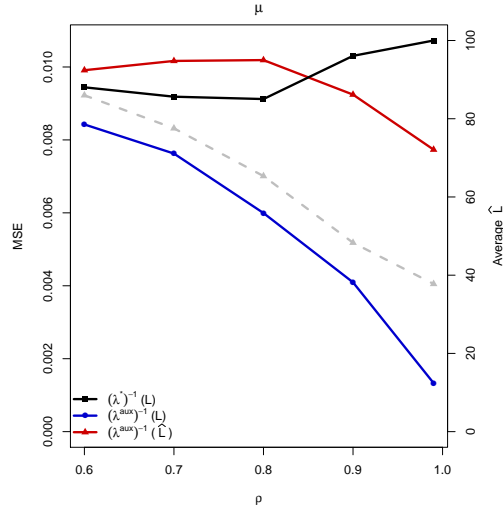


Figure A.6: Plot of the MSE of  $(\hat{\lambda}^*)^{-1}(L)$ ,  $(\hat{\lambda}^{aux})^{-1}(L)$ , and  $(\hat{\lambda}^{aux})^{-1}(\hat{L})$  for estimating  $\mu$  across increasing  $\rho$  where  $L = 100$  and  $C_x = .05$ . Also included is the average computational cost,  $\hat{L}$ , used when computing  $(\hat{\lambda}^{aux})^{-1}(\hat{L})$  (dashed line).

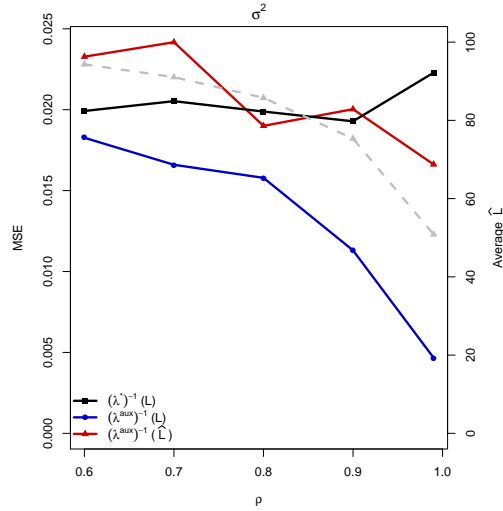


Figure A.7: Plot of the MSE of  $(\hat{\lambda}^*)^{-1}(L)$ ,  $(\hat{\lambda}^{aux})^{-1}(L)$ , and  $(\hat{\lambda}^{aux})^{-1}(\hat{L})$  for estimating  $\sigma^2$  across increasing  $\rho$  where  $L = 100$  and  $C_x = .1$ . Also included is the average computational cost,  $\hat{L}$ , used when computing  $(\hat{\lambda}^{aux})^{-1}(\hat{L})$  (dashed line).

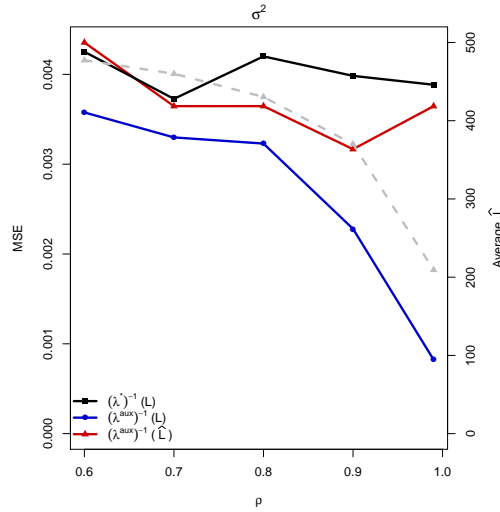


Figure A.8: Plot of the MSE of  $(\hat{\lambda}^*)^{-1}(L)$ ,  $(\hat{\lambda}^{aux})^{-1}(L)$ , and  $(\hat{\lambda}^{aux})^{-1}(\hat{L})$  for estimating  $\sigma^2$  across increasing  $\rho$  where  $L = 500$  and  $C_x = .1$ . Also included is the average computational cost,  $\hat{L}$ , used when computing  $(\hat{\lambda}^{aux})^{-1}(\hat{L})$  (dashed line).

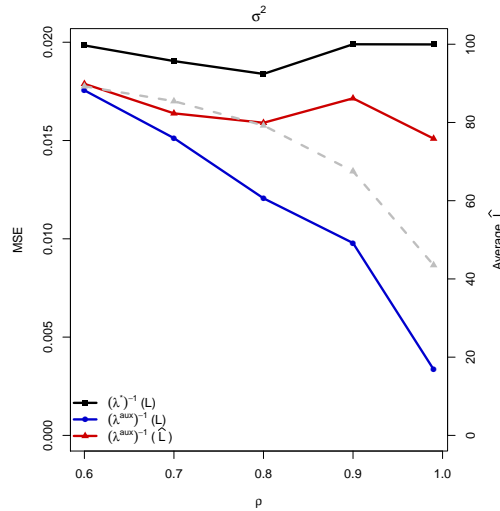


Figure A.9: Plot of the MSE of  $(\hat{\lambda}^*)^{-1}(L)$ ,  $(\hat{\lambda}^{aux})^{-1}(L)$ , and  $(\hat{\lambda}^{aux})^{-1}(\hat{L})$  for estimating  $\sigma^2$  across increasing  $\rho$  where  $L = 100$  and  $C_x = .05$ . Also included is the average computational cost,  $\hat{L}$ , used when computing  $(\hat{\lambda}^{aux})^{-1}(\hat{L})$  (dashed line).