ABSTRACT

XU, STEVEN GUANXING. Advances in Semiparametric Quantile Regression. (Under the direction of Brian Reich and Shu Yang).

Quantile regression (QR) has become an indispensable tool in statistical research for its many advantages over least-squares mean regression methods. It imposes less restriction on the error distribution and allows a more nuanced characterization of the stochastic relationship between covariates and outcome. Furthermore, a set of regression quantiles can often provide a more comprehensive description of the outcome's distribution than a single estimate of the central tendency. Traditional QR is concerned with parametric estimation in which the conditional quantile is often assumed to be a linear combination of the covariates. However, the strict linearity assumption lacks flexibility and cannot accurately characterize highly nonlinear outcome-covariates relationships that are naturally present in economics, health and environment studies. Semiparametric QR overcomes this limitation by unrestricting the shape of the estimated quantile curve. However, many problems that concern QR become substantially more challenging as a trade-off for the flexibility. This dissertation focuses on proposing novel solutions for some of the well-known problems in the QR literature using semiparametric approaches, as well as investigating the extensions and applications of these approaches under different statistical settings.

First, we consider the task of estimating multiple regression quantiles under noncrossing constraints. QR methods model each regression quantile individually. When an investigator wishes to apply such methods to multiple quantiles, however, the separately estimated quantile curves may cross. Quantile crossing violates basic probability rules, since any valid conditional quantile function should be monotonically non-decreasing in the quantile level. Such problem is more severe in nonparametric QR due to their overwhelming flexibility. Even if crossing is not a concern, unconstrained QR significantly inflates the estimation uncertainty in regions where data is scarce. In Chapter 2, we propose a Bayesian nonparametric method that leverages the flexibility of spline approximation and neural network to simultaneously estimate noncrossing, nonlinear quantile curves. Simulation studies show that our model possesses appealing advantages over existing methods, and can better recover quantiles of the response distribution when the data is sparse. Statistical computing is an important aspect of empirical research, and well-developed open-source implementation allows existing methodology to attract a wider usership. As such, Chapter 3

is dedicated to the presentation of a fully-loaded R package that implements the semi/nonparametric QR model proposed in Chapter 2. By using state-of-the-art optimization routine for deep learning, the proposed implementation generalizes the estimation framework in Chapter 2, and provides a scalable frequentist algorithm that excels at computation efficiency in addition to the fully Bayesian algorithm that allows uncertainty quantification. We illustrate the available functionalities of the package in detail using both simulated and real-world data.

Next, we consider the task of estimating causal quantile effects in presence of many confounders. Statistical methods for estimating causal effects of a treatment or exposure using observational data play a central role in many research fields. Standard causal inference characterizes treatment effect through averages, but quantile treatment effects (QTEs) can provide a more comprehensive picture of the treatment effect when the counterfactual distributions differ by not only a location shift. In Chapter 4, we adapt the semiparametric QR model proposed in Chapter 2 to flexibly model the conditional potential outcome distribution, which allows inference on any functionals of counterfactual distributions including probability density functions (PDFs) and multiple QTEs. We propose a double score regression adjustment that augments the propensity score with individual covariates, and develop an approximate Bayesian estimation framework that appropriately propagates modeling uncertainty. We show via simulations that the use of double balancing score for confounding adjustment improves performance over adjusting for any single score alone, and the proposed semiparametric model estimates QTEs more accurately than other semi/nonparametric methods.

Finally, we consider the task of estimating regression quantiles with incomplete observations. Missing data are a frequently encountered problem in many applications, and complete-case (CC) analysis that simply discards all incomplete observations can result in biased and inefficient estimators. Single-index QR models have gained increasing popularity in conditional quantile estimation. They avoid the “curse of dimensionality" by projecting the multivariate covariates to a linear index while retaining flexibility of a nonparametric model through an unspecified link function. Despite abundant literature on QR with missing data, only secant attention has been paid to single-index QR when the data contain missing values. In Chapter 5, we propose a class of weighted pseudo-likelihood estimation procedures that includes inverse probability weighting (IPW), estimating equations projection, and a combination of both. Interestingly, we show, for the first time in the literature, the three approaches correspond to IPW using a parametric model, kernel regression,

and their combination. By using spline approximation and profile likelihood approach, we construct a single objective function that can be optimized by an efficient algorithm. A simulation study shows that, compared to CC estimation, the proposed methods effectively reduced estimation bias when covariates are missing, and stabilize the estimation algorithm in general.

Advances in Semiparametric Quantile Regression

by
Steven Guanxing Xu

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
2022

APPROVED BY:

Brian Reich Co-chair of Advisory Committee	Shu Yang Co-chair of Advisory Committee
Sujit Ghosh	Jacqueline Hughes-Oliver

Ana-Maria Staicu

DEDICATION

To my wife.

BIOGRAPHY

Steven was born in Clovis, California but raised in Beijing, China. At a young age, he visited the U.S. regularly and even finished one semester of elementary school in Cary, North Carolina. He came back to the U.S. before senior year of high school. After graduating from Davis Senior High School, he was admitted to University of Washington, Seattle, where he majored in Statistics and obtained his Bachelor's degree in 2017. He continued to pursue a Ph.D. degree in Statistics at North Carolina State University, where he was fortunate to conduct research under the guidance and direction of Dr. Brian Reich and Dr. Shu Yang.

ACKNOWLEDGEMENTS

First and foremost, I would like to thank my advisors, Dr. Brian Reich and Dr. Shu Yang, whose expertise laid the foundation for my thesis and guidance helped me grow as an independent researcher. You have always been patient, supportive, and willing to answer any silly questions that I had. I would also like to acknowledge other members of my committee. Thank you Dr. Jacqueline Hughes-Oliver for teaching the best introductory graduate courses. Thank you Dr. Sujit Ghosh for teaching the Bayesian course that piqued my interest in Bayesian inference. Thank you Dr. Ana-Maria Staicu for a great semester of advanced statistical inference. Thank you all for your invaluable feedback that helped improve my thesis tremendously.

I would like to thank everyone in the Reich's lab. Your questions and comments helped improve my presentation skill and were determinants of my successful defense. I would like to thank my fellow Ph.D. students Zun Yin, Yiran Wang, Qun Sui, Yang Sun, Yukun Song. Your friendships are priceless and were what kept my monotonous life fulfilling.

I owe many thanks to my wife, whose unconditional support and encouragement helped me stay motivated when facing challenges and bottlenecks. Without you this dissertation would not have been possible. I would also like to thank my parents who insisted that I pursue Ph.D. study which became one of the most memorable years of my life.

Finally, I would like to thank my furry pals, Frito and Lilac, whose companionship always warms my heart and relieves my stress.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

CHAPTER

1

INTRODUCTION

1.1 Problem of interest

Quantile regression (QR) models the statistical relationship between conditional quantiles of the response distribution and a set of covariates using linear or nonlinear regression equation. It has become widely used in diverse areas to complement least-squares regression when investigators are interested in covariate-effect on non-central parts of the response distribution (Koenker 2005). For example, a physician might be interested in modeling the 0.05 conditional quantile of birth-weight distribution to understand the determining factors of underweight newborns (Abrevaya 2001); a climatologist might be interested in modeling the 0.99 conditional quantile of wind speed distribution to study the behavior of tropical cyclones that may cause major damage (Jagger and Elsner 2009). In addition, a set of regression quantiles can offer a more comprehensive description of the conditional response distribution than a single estimate of the central tendency.

Let Y be the response variable, and $\boldsymbol{X}$ be the p-dimensional covariates. For any given quantile level $\tau \in (0,1)$ and $\boldsymbol{x} \in \mathbb{R}^p$, the conditional quantile of Y given $\boldsymbol{X} = \boldsymbol{x}$ is defined as

the solution to the following optimization problem

$$G_\tau(\boldsymbol{x}) \coloneqq Q_Y(\tau|\boldsymbol{X}=\boldsymbol{x}) = \arg\min_a \mathbb{E}\left\{\rho_\tau(Y-a)|\boldsymbol{X}=\boldsymbol{x}\right\},$$

where $\rho_\tau(u) = \tau u - u1(u<0)$ is the check loss function. Traditional QR is concerned with parametric estimation in which the conditional quantile is often assumed to be a linear combination of the covariates:

$$G_\tau(\boldsymbol{x}) = \boldsymbol{x}^\top \boldsymbol{\beta}_\tau,$$

where β_τ is a p-dimensional vector of unknown parameters. However, the strict linearity assumption lacks flexibility. To overcome this limitation, semi/nonparametric QR models that impose minimal restriction on the shape of regression quantiles have been considered in many works; see for example, Yu and Jones (1998), Takeuchi et al. (2006), Dette and Volgushev (2008), Taddy and Kottas (2010), Moon et al. (2021). With great flexibility comes great challenge. Many recurrent estimation problems that have been studied extensively under the assumption of a parametric QR model become substantially more difficult to approach when such restriction is lifted, and therefore have only received scant attention in the semiparametric QR literature.

There is a large body of work on QR. In this dissertation, we identify a selection of estimation problems that have yet receive a satisfactory treatment under a semiparametric estimation framework. Specifically, we focus on three topics, namely noncrossing QR, causal quantile effects estimation, and QR with incomplete data. For each topic, we start off by reviewing the classical approaches as well as recent advancements, and identify apparent shortcomings associated with existing methods that requires improvement. We then propose a novel solution to address at least some of the shortcomings and provide empirical evidence to justify its advantage over existing methods. Finally, we discuss possible future research directions to tackle the remaining issues and improve the proposal from different aspects.

1.2 Literature review

1.2.1 Noncrossing quantile regression

Arguably, the full potential of QR lies in the simultaneous description of a(n) (infinite) collection of regression quantiles

$$\{G_\tau(\boldsymbol{x}),\ 0<\tau<1\},$$

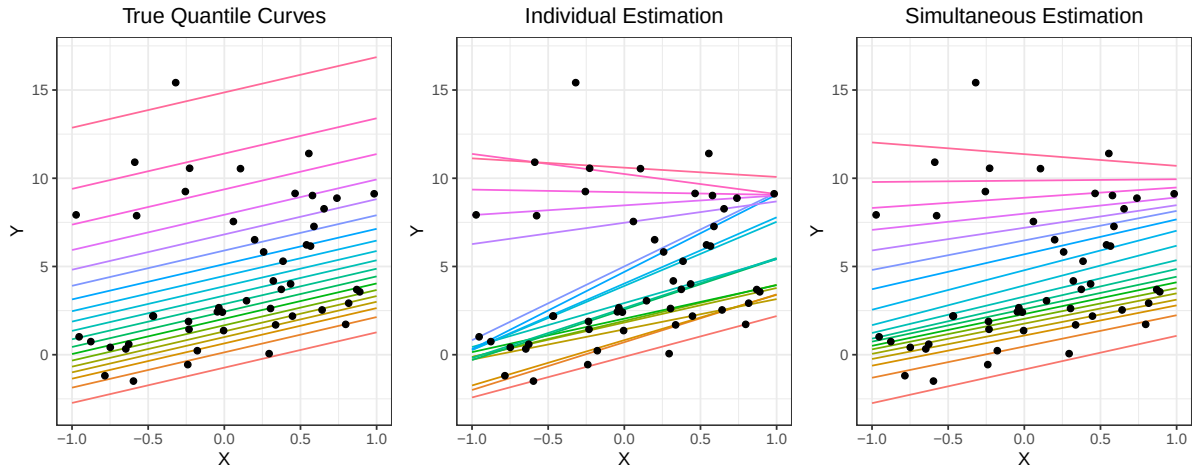
which is of particular interest in many applications when the investigator wants to monitor how the effect of a covariate change across the quantile domain. A naive solution to estimate multiple regression quantiles is to fit separate QR models to each quantile level of interest. However, because the different regression equations are solved independently, the correlation structure of the true conditional quantiles will not be retained. Furthermore, because no restriction on the independent estimates were placed *a prior*, separately fitted QRs will not take into account the natural ordering among different quantiles and will lead to estimated quantile curves of which two or more might cross (see Fig. 1.1 for illustration). Quantile crossing persists and is more severe under a semi/nonparametric regression setting, as the individually fitted curves become overly flexible and can easily overfit the data when the sample size is small.

Existing methods that alleviate quantile crossing can be roughly classified into three categories: sequential estimation, post-processing and simultaneous estimation. In sequential estimation, different quantile curves are estimated sequentially under the constraint that the currently estimated curve should not cross the previous one (Liu and Wu 2009; Muggeo et al. 2013). However, an obvious drawback is that each estimated quantile only borrows information from its preceding estimate, but the former might also contain information that would have been helpful in estimating the latter.

In post-processing, some adjustment are applied on the unconstrained QR estimates so that the monotonicity of the predicted conditional quantile function is enforced (Dette and Volgushev 2008; Chernozhukov et al. 2010; Rodrigues and Fan 2017). However, the performance of the final estimator depends on the initial estimates, which could be poor as they do not borrow information from each other.

Finally, in simultaneous estimation, regression quantiles for a set of quantile levels are estimated jointly under the constraint that no two of them will cross each other. This is the

Figure 1.1: Comparison of individually and simultaneously estimated quantile curves. The data are generated from $Y = 1 + 2X + \epsilon$ with $n = 50$, $X \sim \mathcal{U}(-1, 1)$ and $\epsilon \sim ALD(\mu = 0, \sigma = 1, p = 0.2)$. Individual and simultaneous estimates are calculated using the method of Koenker and Bassett Jr (1978) and Yang and Tokdar (2017) respectively. Crossing is severe when quantile curves are estimated individually but is alleviated when they are estimated simultaneously. Simultaneous estimation also leads to significant improvement in overall precision by borrowing information across adjacent quantiles.



most appealing approach of the three as it directly incorporates the noncrossing constraint in modeling and requires optimizing only one objective function. Under simultaneous estimation, each regression quantile is somewhat regularized by its adjacent estimates, which can especially improve overall performance when the data size is small.

Nonlinear simultaneous QR models are a minority in the QR literature, and existing approaches suffer from apparent shortcomings. Cannon (2018) proposed to model the quantile process using a feed-forward neural network. They consider the quantile level as a regressor taking fixed values, and impose partial monotonicity constraint on its weight coefficients to enforce monotonicity of the quantile function. Although their approach elegantly avoids quantile crossing, additional quantiles outside the pre-specified range have to be estimated via extrapolation. Das and Ghosal (2018) model the quantile process as a weighted sum of B-spline basis functions of the quantile level; the weights are further expanded by tensor products of B-spline series expansion of each covariate, and order constraints are imposed on the spline coefficients to ensure non-crossing. Although their method models the full quantile process, it does not scale well to high dimensions since the number of parameters grows exponentially with the number of covariates. Nonlinear

quantile process can also be estimated by inverting (or integrating then inverting) any valid estimate of the conditional distribution function. Das and Ghoshal (2018) proposed a model on the conditional cumulative distribution function (CDF) of similar form to their aforementioned quantile process model. Consequently, their CDF model also suffers from computational intractability in high dimensions. Furthermore, their model is not constrained properly to estimate a bona fide density, which often leads to poor numerical performance. Izbicki and Lee (2016) projected the conditional probability density function (PDF) onto data-dependent eigenfunctions of a kernel-based operator. Their model scales well to high dimensions and estimates a bona fide density, but the resulting quantile surfaces are often not smooth.

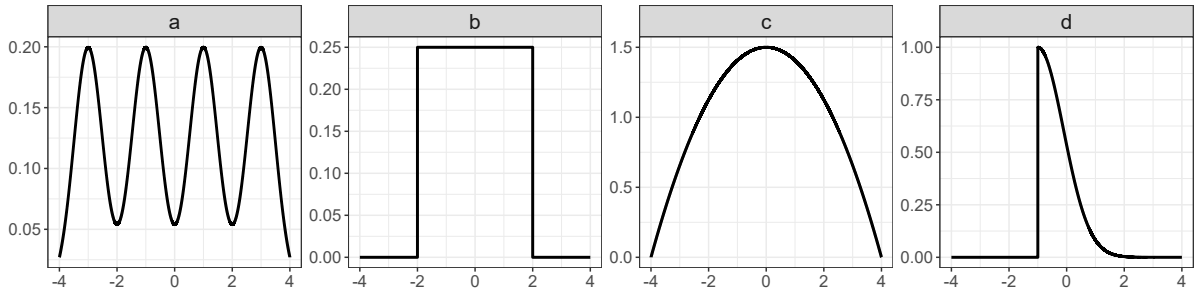
1.2.2 Causal quantile effects estimation

Causal inference is the study of how interventions or treatments affect outcomes of interest. It plays an important role in many fields. In social and economic studies, researchers are interested in questions such as “Will job training program improve real earnings?" (Imbens and Rubin 2015). In environmental studies, researchers may ask “Will eutrophication cause heavy metal pollution?"(Sun et al. 2021). Observational studies present barriers for estimation of treatment effect since baseline characteristics of treated subjects can differ greatly from those of untreated subjects. Most of the existing methods are cast in terms of the potential outcomes framework (Rubin 1978), which commonly assumes strong ignorability of treatment assignment (i.e., conditionally independent of the potential outcomes given the confounders). Under the strong ignorability assumption, causal effects can be estimated using the observed outcome after appropriate adjustment for confounders.

Classical causal literature often quantify the causal effects with mean. The average treatment effect (ATE), for example, measures the difference in mean outcome had all versus none in a population been treated. However, when counterfactual distributions under treatment and control differ in not only central tendency, ATE might be insufficient or even fails to reveal the distributional differences. To illustrate, consider the densities in Fig. 1.2, which all have exactly the same mean. These distributions are obviously different, but will be indistinguishable with the ATE.

Quantile treatment effects (QTE) (Doksum 1974) measure the differences between quantiles of potential outcome. It can capture heterogeneous causal effects of the treatment at different locations of the counterfactual distribution. Counterfactual quantiles can be

Figure 1.2: Densities of four distributions, all with mean 0.



estimated by either minimizing a weighted check loss or inverting an estimate of the cumulative distribution function (CDF) of the potential outcomes. In the latter case, the CDF can be naturally estimated using empirical expectation of thresholded outcomes, and therefore estimation of counterfactual quantiles largely reduces to estimation of counterfactual mean (e.g., Firpo 2007; Rothe 2010; Zhang et al. 2012; Donald and Hsu 2014; Yang and Zhang 2020; Sun et al. 2021). However, since independent objective functions/estimating equations are optimized/solved for different quantile levels, these methods share the same drawback of separate QR discussed in 1.2.1. In addition, the discrete nature of these estimators forbids inference on the counterfactual probability density function (PDF). The PDF can reveal potentially interesting characteristics of the counterfactual distribution that cannot be revealed by the CDF, such as multimodality, and are often more visually interpretable to practitioners.

Counterfactual PDF estimation is a statistically more challenging problem than counterfactual CDF (or equivalently, quantile function) estimation. This is also true in the non-counterfactual setup, where the CDF can be estimated with empirical means at $n^{-1/2}$ rates, while the PDF requires a more careful estimation procedure to balance between smoothness and variance. Some early attempts have discussed possible estimators (DiNardo et al. 1996; Robins and Rotnitzky 2001) for counterfactual densities, but none of them proceed in full analysis. Recently, Kim et al. (2018) adopted the kernel estimator from Robins and Rotnitzky (2001) and proposed a doubly robust like estimator that uses inverse probability weighting (IPW) for bias correction, but the resulting estimate is highly non-smooth. This limitation is subsequently alleviated by Kennedy et al. (2021) who projects the nonparametric PDF to a truncated cosine series whose oscillating nature, unfortunately, may cause poor boundary estimation.

Semi/nonparametric estimation of counterfactual distribution and its associated causal effects has also received growing interest from the Bayesian nonparametric (BNP) literature. A representative work is that of Xu et al. (2018) who combined propensity score (PS) adjustment and BNP conditional density estimation. First, the PS is modeled using probit regression and Bayesian additive regression trees (BART; Chipman et al. 2010). For each treatment group, the counterfactual distribution conditional on the estimated PS is then modeled using a Dirichlet process mixture (DPM) of normals. Although PS adjustment is attractive for dimension reduction purpose, it alone may not summarize the outcome model well and thus cannot sufficiently control the variability of outcome residuals.

1.2.3 Quantile regression with incomplete data

In almost any research or application, there is the potential for missing or incomplete data. Missing data can occur for many reasons, such as non-response, loss to follow up, competing risk, data entry error, data prepossessing (e.g., outlier removal), etc. Many classical and modern statistical techniques require data to be complete, and they simply discard the incomplete records before proceeding estimation otherwise. However, when the data is missing at random (MAR; meaning that the missingness mechanism depends on the observed data but not on the unobserved data), complete-case (CC) analysis that neglects missingness can distort the underlying relationship between covariates and response, resulting in biased and inefficient estimators.

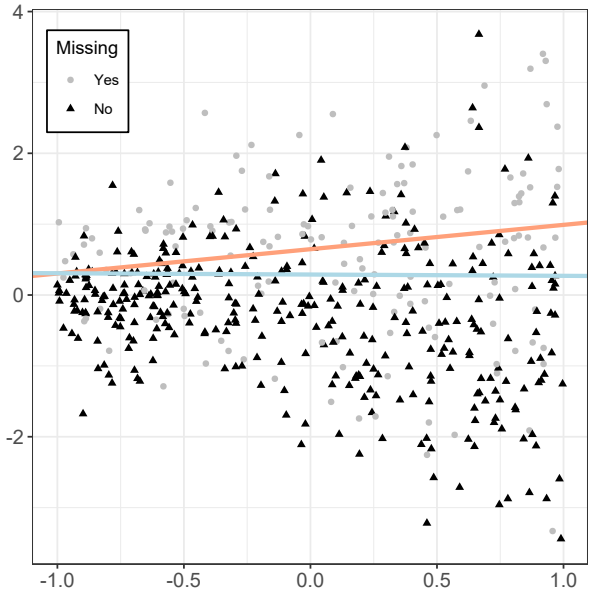
To illustrate, consider estimating the linear regression quantile when the covariates is MAR. Let δ_i be an indicator variable such that $\delta_i = 1$ if X_i is observed and $\delta_i = 0$ otherwise. Under the MAR assumption, the missing probability mechanism is a function of Y_i:

$$P(\delta_i = 1|X_i, Y_i) = P(\delta_i = 1|Y_i).$$

If we ignore missingness and adopt a CC analysis, then we essentially assume a working distribution $Y|X, \delta = 1$, which is different from $Y|X$ due to the confounding effect of δ. Fig. 1.3 compares the estimated regression quantile using the full data and the CC data. The CC estimator significantly underestimates the true slope, providing a misleading description of the quantile-covariate relationship.

Imputation and inverse probability weighting (IPW) are two widely adopted approaches for handling missing data. The former replaces the missing variables in the objective func-

Figure 1.3: Effect of missing data on regression quantile estimation. The data are generated from $Q_Y(\tau|X) = 2(\tau - 0.5)X + \Phi^{-1}(\tau)$, and X is MAR given Y. The orange and blue lines correspond to the 75th regression quantile estimated using full and complete-case (CC) data, respectivefully.



tion/estimating equation with their imputed values, whereas the latter reweights the CC objective function/estimating equation by inverse PS. In this dissertation, we focus on the weighting approach.

Among works that apply IPW to estimate regression quantiles with MAR data, Sherwood et al. (2013) and Chen et al. (2015) addressed the case of linear QR, while Zhou et al. (2021) tackled the case of nonparametric QR using local linear regression techniques (Fan and Gijbels 1992). PS is often modeled using a parametric model to avoid “curse of dimensionality", but parametric IPW approach is prone to model misspecfication. To overcome this limitation, estimating equation projection (EEP; Zhou et al. 2008) and augmented IPW (AIPW) approaches that leverage the efficiency of a nonparametric kernel estimator have been proposed. Their advantages over IPW approach have been investigate by Chen et al. (2015) under linear QR, and Wang et al. (2022) under nonparametric QR. One unsatisfactory feature of these works is that they are all concerned with parametric or fully nonparametric estimation of regression quantiles, and an middle ground solution based on semiparametric estimation is currently missing in the literature.

Single-index models (Ichimura 1993; Hardle et al. 1993) have gained increasing pop-

ularity in conditional quantile estimation. They avoid the “curse of dimensionality" by projecting the multivariate covariates to a linear index while retaining flexibility of a nonparametric model through an unspecified link function. Despite abundant literature on QR with missing data, only scant attention has been paid to single-index QR when the data contain missing values. Zou et al. (2020) defined weighted estimators of index parameters and link function for partially linear single-index QR when the response is censored and the censoring indicator is MAR. Liang et al. (2021) focused on the general MAR data setting, and Liu and Liang (2022) developed a Bayesian framework. These methods all focused on the parametric IPW method which is prone to model misspecification.

1.3 Organization

The rest of the dissertation is organized as follows. In Chapter 2, we propose a nonparametric modeling framework for the conditional response distribution to simultanesouly estimate all possible regression quantiles while ensuring they do not cross. We further address issues with uncertainty quantification and model interpretation by detailing a fully-Bayesian estimation framework and extending model-agnostic tools designed for mean regression to applications regarding QR. In Chapter 3, we address issues with statistical computation by proposing a novel R package that implements the model proposed in Chapter 2. We propose a scalable frequentist algorithm that leverages GPU capability and is capable of estimating “deep" regression quantiles.

In Chapter 4, we extend the semiparametric QR mooel proposed in Chapter 2 to the potential outcome framework for estimating counterfactual distributions in presence of many confounders. We propose the use of double balancing score regression adjustment to aggregate information from both treatment assignment mechanism and individual covariates for more efficient estimation of the quantile treatment effects. We also address issues with uncertainty quantification by outlining an approximate Bayesian estimation framework.

In Chapter 5, we shift our focus to single-index QR and estimation of its parameters with missing at random data. We propose a class of weighted pseudo-likelihood estimation procedures that summarize inverse probability weighting (IPW), estimating equation projection (EEP), and augmented IPW (AIPW) approaches. We provide theoretical evidence that the three approaches are more profoundly connected than structural resemblance,

and demonstrate through numerical studies the bias-reduction advantage of proposed approaches over complete-case (CC) estimation.

The related work and our unique contributions are discussed thoroughly in these chapters, and numerical results are presented to support the proposed methods. Supplemental information including computation detail and additional results for each chapter are relegated to the Appendix sections.

CHAPTER

2

A BAYESIAN NONPARAMETRIC MODEL FOR NONCROSSING QUANTILE REGRESSION

The work in this Chapter has been published in *Biometrics* with a paper entitled “Bayesian Non-parametric Quantile Process Regression and Estimation of Marginal Quantile Effects" (Xu and Reich 2021).

2.1 Background

Traditional quantile regression (QR) models treat estimation of different quantiles as independent problems. When these models are fitted separately for inference on multiple levels, the natural ordering among different quantiles cannot be ensured, and the estimated quantiles are subject to cross. Quantile crossing can be alleviated by solving a constrained optimization problem (Bondell et al. 2010; Liu and Wu 2011) when only a grid of quantiles

is modeled, but estimates based on these methods can be sensitive to the number and location of the chosen quantile grids.

Simultaneous QR (SQR) allows inference on all quantiles by specifying the full quantile process. It encourages strength borrowing across proximate quantile levels through an unified modeling approach. SQR was first proposed by He (1997) who assumes a linear heteroscedastic regression model for the response. Subsequently, linear SQR models that impose fewer restrictions on the quantile function have been developed (e.g., Reich and Smith 2013; Yuan et al. 2017; Yang and Tokdar 2017). These approaches enjoy great interpretability by allowing rate-of-change interpretation of quantile-dependent coefficients but cannot accommodate quantile curves with complex nonlinear trends. Furthermore, they are not suitable for high-dimensional problems since they do not implicitly account for interaction effects.

Nonlinear SQR models are a minority in the current quantile regression literature, and existing approaches suffer from apparent shortcomings. Cannon (2018) proposed to model the quantile process using a feed-forward neural network. They pre-specify a set of quantile levels and treat them as a monotone covariate in the model to enforce monotonicity of the quantile function. Although their approach elegantly avoids quantile crossing, additional quantiles outside the pre-specified range have to be estimated via extrapolation. Das and Ghosal (2018) model the quantile process as a weighted sum of B-spline basis functions of the quantile level; the weights are further expanded by tensor products of B-spline series expansion of each covariate, and order constraints are imposed on the spline coefficients to ensure non-crossing. Although their method models the full quantile process, it does not scale well to high dimensions since the number of parameters grows exponentially with the number of covariates. Nonlinear quantile process can also be estimated by inverting (or integrating then inverting) any valid estimate of the conditional distribution function. Das and Ghoshal (2018) proposed a model on the conditional cumulative distribution function (CDF) of similar form to their aforementioned quantile process model. Consequently, their CDF model also suffers from computational intractability in high dimensions. Furthermore, their model is not constrained properly to estimate a bona fide density, which often leads to poor numerical performance. Izbicki and Lee (2016) projected the conditional probability density function (PDF) onto data-dependent eigenfunctions of a kernel-based operator. Their model scales well to high dimensions and estimates a bona fide density, but the resulting quantile surfaces are often not smooth. Recently, SQR models that leverage the advancement of deep learning have also been developed (e.g., Kim et al. 2021), but the

primary focus of these models is on prediction rather than inference.

2.2 Contribution

In this chapter, we propose a novel treatment to nonlinear SQR by specifying a Bayesian nonparametric model on the conditional distribution. While there exist other conditional distribution regression models (e.g., Holmes et al. 2012; Li et al. 2021), we model the conditional CDF using an I-spline basis expansion; the spline coefficients are modeled as functions of covariates using neural networks with specific output activation functions that ensure the model represent a bona fide CDF. We choose to model the distribution function instead of the quantile process because the former permits analytic derivation of the likelihood function and therefore efficient MCMC sampling of the posterior, and for a nonparametric regression the span of potential models is the same in both cases. A spline-based model ensures the estimated conditional CDF, and therefore the estimated conditional quantile function is smooth, and the neural networks allow incorporation of complex covariate effects on the response distribution. We name this method "QR Using I-spline Neural Network (QUINN)". QUINN provides several improvements over existing nonlinear SQR models (Izbicki and Lee 2016; Cannon 2018; Das and Ghosal 2018). Instead of treating the quantile levels as a monotone covariate, QUINN specifies the full CDF such that all quantiles, rather than only a subset, can be modeled without extrapolation. By imposing proper constraints on the spline coefficient functions, we ensure that the estimate is a bona fide CDF, leading to more accurate estimation of the quantile process; directly constraining the model also avoids the need of an post hoc normalization method which often renders the final quantile estimates unsmooth. Finally, by expanding the covariate-dependent coefficient functions using FNN rather than tensor products of splines, we greatly reduce the dimensionality of the parameter space so that it only scales linearly, instead of exponentially, in the number of covariates. A relevant but distinct work is that of Smith et al. (2015) who proposed a semi-parametric framework based on I-spline basis expansion for a simultaneous estimation of linear quantile planes. In contrast, QUINN models the conditional distribution nonparametrically and allows simultaneous estimation of arbitrary quantile surfaces.

A disadvantage of modeling the conditional CDF nonparametrically is that covariate effects on different quantiles are not self-explanatory. This is common for black box super-

vised learning models that sacrifice transparency for flexibility. To overcome this challenge, model-agnostic methods (Ribeiro et al. 2016) have been developed to extract interpretation from any supervise learning model. A recent contribution is made by Apley and Zhu (2020) who proposed accumulated local effect (ALE) plot to visualize main and second-order interaction effects of a black box supervised learning model. Their method produces reliable characterization of the covariate effects on the predicted response in a computationally efficient way. In this chapter, we show that ALE plots can be applied to visualize covariate effects on predicted quantiles. We also present ways to estimate feature importance of marginal quantile effects of QUINN.

The motivating example is a study analyzing the effect of pregnancy related and demographic factors on the distribution of birth outcomes. Low birth weight (LBW) is defined as weight less than 2.5kg. It is a leading cause of prenatal and neonatal deaths, and births of underweight infants result in long-term medical and economic costs. High birth weight (HBW) is defined as weight greater than 4kg. It is also an emerging public health issue worldwide. Overweight infants are subject to increased risk of health problems after birth, such as obesity in early childhood. QR is a natural approach to understand the determinants of LBW and HBW by modeling the lower and upper quantiles of the birth weight distribution. Examples are the separate QR approach by Abrevaya (2001) and SQR approach by Tokdar et al. (2012). These works all assume a linear regression model, which will mischaracterize effects that are nonlinear (Ngwira and Stanley 2015). In this chapter, we apply QUINN to the 2019 U.S. Natality Data Set (National Center for Health Statistics 2019) to flexibly model different quantiles of the birth weight distribution, with a primary focus on identifying the influential factors of LBW and HBW.

2.3 Organization

The rest of the chapter is organized as follows. In Section 2.4 we introduce the nonparametric density regression model that can be used for simultaneous quantile estimation. In Section 2.5 we briefly review ALE plots and explain how it can be used to extract interpretation from the proposed model. In Section 2.6 we compare the proposed model to existing nonparametric simultaneous QR models through a simulation study. In Section 2.7 we analyze the quantile effects of pregnancy-related factors on birth weight. We conclude the chapter with a discussion in Section 2.8.

2.4 Methodology

Denote $\boldsymbol{X} = (X_1, ..., X_d) \in \mathscr{X}$ as the covariate vector and Y as the scalar response. We are interested in approximating the quantile process of the response given the covariates $Q_Y(\tau|\boldsymbol{X} = \boldsymbol{x})$ for quantile level $\tau \in (0,1)$ and all $\boldsymbol{x} \in \mathscr{X}$. If $Q_Y(\tau|\boldsymbol{x})$ is continuous and monotonically increasing in τ, then for any $\boldsymbol{x} \in \mathscr{X}$ the conditional CDF is $F_Y(y|\boldsymbol{x}) = Q_Y^{-1}(\tau|\boldsymbol{x})$. Thus, the monotonicity constraint of $Q_Y(\tau|\boldsymbol{x})$ can be naturally accounted for by specifying a valid model on $F_Y(y|\boldsymbol{x})$ and then inverting it. Our method requires the response variable to have a lower and upper bound, which we achieve by introducing a transformed variable $Z = g(Y)$ for some monotonic function g that maps to the unit interval. In this section, we will outline our method for approximating quantile process of the transformed response $Q_Z(\tau, \boldsymbol{x})$ whose estimate can then be back-transformed to an estimate of $Q_Y(\tau|\boldsymbol{x})$.

2.4.1 Density regression using shape-constrained splines

We propose to model the conditional density of Z given $\boldsymbol{x}$ using shape-constrained regression splines. Specifically, we model the conditional PDF using M-splines, and the conditional CDF using I-splines. The M-spline family is a set of piecewise polynomials having properties of non-negativity and unit integral. In other words, each basis function has the properties of a PDF. Let $0 = t_0 < t_1 < \cdots < t_{p+1} = 1$ be a partition of the unit interval with p equally spaced internal knots. Let $\{M_j(u) : 1 \le k \le K\}$ be a set of M-spline basis functions on $[0,1]$ of order r with p internal knots such that $K = r + p$. A convex combination of M-spline basis functions, i.e.,

$$\sum_{k=1}^{K} \theta_k M_k(\cdot) \text{ s.t. } \theta_k \ge 0 \; \forall k \text{ and } \sum_{k=1}^{K} \theta_k = 1,$$

is a valid model for a PDF with support on $[0,1]$. The shape of the modeled PDF can be further controlled by placing additional constraints on the coefficients θ_k. For example, setting $\theta_K = 0$ will force the M-spline to return to 0 at unity.

Similarly, let $\{I_k(u) : 1 \le k \le K\}$ be the set of I-spline basis functions on $[0,1]$ of order r with the same knots. I-splines are defined as the integral of M-splines

$$I_k(x) = \int_0^x M_k(u) d\,u, \; k = 1, ..., K$$

and are piecewise polynomials of degree $r+1$. Since M-splines are non-negative and integrate to 1, I-splines are monotonically non-decreasing with range $I_k(0)=0$ and $I_k(1)=1$ for all j. Thus, a convex combination of I-spline basis functions, i.e.,

$$\sum_{k=1}^{K} \theta_k I_k(\cdot) \text{ s.t. } \theta_k \geq 0 \; \forall k \text{ and } \sum_{k=1}^{K} \theta_k = 1, \tag{2.1}$$

is a valid model for a CDF with support on the unit interval.

The use of shape-constrained regression splines offers an attractive solution to density estimation problems. Many theoretical works have shown the approximation power of non-negative splines and monotone splines. For example, Beatson (1982) shows that as the number of knots increases, the space of non-negative splines converges to the space of non-negative continuous functions almost as quickly as unconstrained splines. Chui et al. (1980) show an analogous result for monotonic splines on approximating continuous monotonic functions. Through numerical studies, Abrahamowicz et al. (1992) show that the asymptotic theories are not affected by the addition of simplex constraint, and that M- and I-splines yield satisfactory accuracy in density regression.

2.4.2 QR using I-splines and neural network (QUINN)

Let $F_Z(z|\boldsymbol{x})$ denote the conditional CDF of the transformed response variable Z given $\boldsymbol{x}$. Following (2.1), a flexible model for $F_Z(z|\boldsymbol{x})$ can be expressed as

$$F_Z(z|\boldsymbol{x},\mathscr{W}) = \sum_{k=1}^{K} \theta_k(\boldsymbol{x},\mathscr{W}) I_k(z) \text{ s.t. } \theta_k(\boldsymbol{x},\mathscr{W}) \geq 0 \; \forall k \text{ and } \sum_{k=1}^{K} \theta_k(\boldsymbol{x}) = 1$$

where the covariates affect the conditional CDF through the spline coefficient functions $\theta_k(\boldsymbol{x},\mathscr{W})$ parametrized by $\mathscr{W}$. The coefficient functions govern the covariate effect on the conditional CDF and therefore should be flexible enough to capture complex nonlinear trends and allow for high-order interaction effects. They also need to be properly constrained so that $F_Z(z|\boldsymbol{x},\mathscr{W})$ has the properties of a valid CDF. To satisfy these two requirements, we model $\theta_k(\boldsymbol{x},\mathscr{W})$ using a feed-forward neural network (FNN) with softmax output

activation,

$$\theta_k(\boldsymbol{x}, \mathscr{W}) = \frac{\exp\{u_k(\boldsymbol{x}, \mathscr{W})\}}{\sum_{i=1}^{K} \exp\{u_i(\boldsymbol{x}, \mathscr{W})\}},$$
$$u_k(\boldsymbol{x}, \mathscr{W}) = W_{2m0} + \sum_{l=1}^{V} W_{2ml}\phi\left(W_{1l0} + \sum_{j=1}^{d} W_{1lj}x_j\right),$$

where $\mathscr{W} = \{W_{uvw}\}$ are the unknown weights and ϕ is the known activation function. Throughout this chapter, ϕ is taken to be the hyperbolic tangent function. Profiting from its universal approximation theorem (Hornik et al. 1989), FNN allows the unconstrained coefficient functions $u_k(\boldsymbol{x}, \mathscr{W})$ to describe arbitrarly complex covariate effects. While the softmax activation naturally projects $u_k(\boldsymbol{x}, \mathscr{W})$ to the unit simplex and overcomes the challenge of parameter estimation under monotonicity constraints. For simplicity, we describe the FNN with a single hidden layer with V neurons, but extensions to deeper networks are straightforward.

The proposed model can approximate any continuous conditional CDF. Following the results of Chui et al. (1980) and Abrahamowicz et al. (1992), with a large enough p, we can assume for any $\boldsymbol{x}$ there exists a set of non-negative coefficients $\{\alpha_1, \alpha_2, ..., \alpha_K\}$ satisfying the constraint $\sum_{k=1}^{K} \alpha_k = 1$ such that $\sum_{k=1}^{K} \alpha_k I_k(z)$ approximates the conditional CDF $F(z|\boldsymbol{x})$ arbitrarily well. In QUINN, the mapping $\psi : \boldsymbol{x} \to \{\alpha_1, \alpha_2, ..., \alpha_K\}$ is modeled by a single-hidden-layer FNN with softmax output which is a non-constant, bounded, and continuous function. Then by the universal approximation theorem (Hornik et al. 1989), there exist weights $\mathscr{W}$ such that the single-hidden-layer FNN $\theta_k(\boldsymbol{x}, \mathscr{W})$ approximates the mapping ψ arbitrarily well for all $\boldsymbol{x}$, provided that the number of hidden neurons V is large enough. Thus, by leveraging the approximation power of I-splines and FNN, the model $\sum_{k=1}^{K} \theta_k(\boldsymbol{x}, \mathscr{W}) I_k(z)$ can approximate any conditional CDF $F(z|\boldsymbol{x})$.

We adopt a Bayesian framework to estimate the weights $\mathscr{W}$ by assigning them prior distributions. Compared to its frequentist counterpart, Bayesian neural network modeling captures uncertainty in both the fitted model and weight parameters, and avoids over-fitting when the sample size is small. Zero-mean Gaussian distributions are the most commonly used prior on weights and have been explored in many classic works (MacKay 1992; Neal 1993). Their popularity arise from their "weight-decay" regularization effect which prevents individual nodes from having extreme value. For QUINN, we set $W_{1vw} \overset{indep}{\sim} \mathcal{N}(0, \sigma_w^2)$, $W_{2vw} \overset{indep}{\sim} \mathcal{N}(0, \gamma^2)$ so that weights in input-hidden layer have feature-wise variances, and

weights in hidden-output layer share a common variance. The scale hyperparameters σ_w and γ are also treated as unknown and assigned hyperpriors, so that their values can be optimized by the data. Gelman et al. (2006) recommends half-t families with a small degrees of freedom. These distributions allow the variance to be arbitrarily close to 0 which regularizes the complexity of the model. In practice however, the heavy-tailedness of half-t distributions make them too broad and often cause difficulty in convergence. Therefore we set $\sigma_w, \gamma \overset{iid}{\sim} \mathcal{N}^+(0, a^2)$ to follow the half-Gaussian distribution for simpler posterior geometry. The variance of half-Gaussian prior is set to be $a^2 = 900$ so it is still relatively noninformative. Experiments show that our model is not sensitive to moderately large values of a. The likelihood function of QUINN has a closed-form expression, therefore MCMC algorithms can be used to explore the posterior. However, traditional methods such as random-walk Metropolis and Gibbs sampler do not scale well to high-dimensional posterior with complex geometry. In this chapter, we use No-U-Turn sampler (NUTS) (Hoffman and Gelman 2014) that uses gradient information to sample efficiently from high-dimensional posterior. Appendix A.1 describes the MCMC algorithm used to approximate the posterior.

Our ultimate goal is to estimate the quantile process of the original response variable $Q_Y(\tau|\boldsymbol{x})$. Let $\hat{F}_Z(z|\boldsymbol{x})$ denote the conditional CDF estimator and $\boldsymbol{D}_Z = \{\tilde{z}_1, \tilde{z}_2, \ldots, \tilde{z}_N\}$ denote a dense grid on the unit interval. Nonparametric estimate of the quantile process $Q_Z(\tau|\boldsymbol{x})$ can be easily obtained by first evaluating $\hat{F}_Z(z|\boldsymbol{x})$ on $\boldsymbol{D}_Z$ and then performing linear interpolation on a dense percentile grid by treating $\{\hat{F}_Z(z_i|\boldsymbol{x})\}_{i=1}^N$ as the input values and $\boldsymbol{D}_Z$ as the functional output values. Because of the one-to-one correspondence between quantile function and CDF, the resulting quantile process estimator will also inherit the approximator property of the proposed CDF estimator. Finally, the estimated quantile process of the original response is given by $\hat{Q}_Y(\tau|\boldsymbol{x}) = g^{-1}\left[\hat{Q}_Z(\tau|\boldsymbol{x})\right]$.

As discussed in Section 1, the proposed model has several advantages over existing nonlinear SQR models (Izbicki and Lee 2016; Cannon 2018; Das and Ghosal 2018). The combination of I-splines and FNN leads to a valid probability model that spans a wide class of conditional distribution functions. Also, as described below and shown later by simulation studies, this combination leads to efficient computation and fully-Bayesian inference on quantile effects.

2.5 Summarizing covariate effects

QUINN includes a flexible FNN model for covariate effects across quantile levels. FNN is a “black box" supervised learning model that excel in flexibility but lack transparency. Unlike linear QR models which enable rate-of-change interpretation of the τ-dependent coefficients, the proposed FNN-based model does not characterize the covariate effects on the predicted quantile in a self-explanatory way. This is inconvenient, since QR models are often used for data exploratory purposes.

Fortunately, research on model agnostic methods has allowed post hoc analysis of main effects and second-order interaction effects (we omit consideration of higher order effects as they cannot be visualized or interpreted meaningfully) of the covariates on the predictions made by “black box" models. The most popular model agnostic method is partial dependence plot (PD plot; Friedman 2001) which visualizes the average marginal effect a (pair of) covariate(s) have on the predictions. The PD plot is straightforward to implement and intuitive to interpret, but is expensive to compute. Recently, Apley and Zhu (2020) proposed accumulative local effects (ALEs) plot that provides the same level of interpretation in a more computationally efficient way. In this section, we provide a brief review of the definitions of ALE plot and explain how it can be applied to QR models. We will later demonstrate using a multivariate simulation study how ALE plot can be utilized to extract interpretation from QUINN.

The sensitivity of $Q_Y(\tau|\boldsymbol{x})$ to covariate j is naturally quantified by the derivative $q_j(\tau,\boldsymbol{x}) = \partial Q_Y(\tau|\boldsymbol{x})/\partial x_j$. In a linear QR, the derivative is the scalar effect of covariate j on quantile level τ, but for a nonlinear regression function the derivative depends on $\boldsymbol{x}$. The ALE begins by averaging $q_j(\tau,\boldsymbol{X})$ over $\boldsymbol{X}$ conditioned on $X_j = x_j$, giving $\bar{q}_j(\tau,x_j) = \mathbb{E}_{\boldsymbol{X}}\left[q_j(\tau,\boldsymbol{X})|X_j = x_j\right]$. The uncentered ALE main effect function of X_j is then defined as

$$\bar{Q}_j^U(\tau,x_j) = \int_{x_{\min,j}}^{x_j} \bar{q}_j(\tau,u_j)d\,u_j.$$

The function $\bar{Q}_j^U(\tau,x_j)$ can be interpreted as the ALE of X_j in the sense that it is an accumulation of local effects $\bar{q}_j(\tau,u_j)$ averaged over the distribution of $\boldsymbol{X}$. The uncentered ALE effect does not have a straightforward interpretation because the derivative is invariant to scalar addition, which leads to the definition of the (centered) ALE main effect function $\bar{Q}_j(\tau,x_j)$ that is the same as $\bar{Q}_j^U(\tau,x_j)$ except centered to have mean 0 with respect to the

marginal distribution of X_j.

Analogous formulas define the second-order ALE for X_j and X_l. Consider the second-order partial derivative $q_{jl}(\tau, \boldsymbol{x}) = \partial^2 Q_Y(\tau|\boldsymbol{x})/\partial x_j \partial x_l$. The local effect at $X_j = x_j$ and $X_l = x_l$, averaging over the other covariates, is $\bar{q}_{jl}(\tau, x_j, x_l) = \mathbb{E}_{\boldsymbol{X}}\left[q_{jl}(\tau, \boldsymbol{X})|X_j = x_j, X_l = x_l\right]$. The uncentered second-order ALE is then

$$\bar{Q}^U_{jl}(\tau, x_j, x_l) = \int_{x_{\min,j}}^{x_j} \int_{x_{\min,l}}^{x_l} \bar{q}_{jl}(\tau, u_j, u_l) d\,u_j\, d\,u_l,$$

and the second-order ALE function $\bar{Q}_{jl}(\tau, x_j, x_l)$ is mean-centered with respect to the marginal distribution of (X_j, X_l). The second-order ALE $\bar{Q}_{jl}(\tau, x_j, x_l)$ describes the joint effects of the two covariates, which consist of both their main effects and interaction effect. In cases where assessment of only the interaction effect is of interest, main effects of X_j and X_l can be further subtracted from $\bar{Q}_{jl}(\tau, x_j, x_l)$ to obtain the pure interaction effect $\bar{Q}^I_{jl}(\tau, x_j, x_l)$.

The functions $\bar{Q}_j(\tau, x_j)$, $\bar{Q}_{jl}(\tau, x_j, x_l)$, and $\bar{Q}^I_{jl}(\tau, x_j, x_l)$ can be plotted to understand each main and interaction effect. When plotted against x_j, the main ALE $\bar{Q}_j(\tau, x_j)$ quantifies the difference between average prediction conditioned on $X_j = x_j$ and the average prediction over $\boldsymbol{X}$. When plotted against x_j and x_l, the second-order ALE $\bar{Q}_{jl}(\tau, x_j, x_l)$ quantifies the difference between average prediction conditioned on $(X_j, X_l) = (x_j, x_l)$ and the average prediction over $\boldsymbol{X}$. The interaction ALE $\bar{Q}^I_{jl}(\tau, x_j, x_l)$ can be interpreted analogously to $\bar{Q}_{jl}(\tau, x_j, x_l)$, except now the difference is contributed entirely by the interaction effect.

It is also useful to summarize the main and interaction ALEs with a one-number summary that can be used to rank the importance of each effect. Following Greenwell et al. (2018), we propose to measure overall variable importance (VI) for continuous covariates using the standard deviation of the ALE with respect to the marginal distribution of $\boldsymbol{X}$, i.e., $\text{VI}_j(\tau) = \text{SD}\left[\bar{Q}_j(\tau, X_j)\right]$ and $\text{VI}_{jl}(\tau) = \text{SD}\left[\bar{Q}^I_{jl}(\tau, X_j, X_l)\right]$. For categorical covariates, the standard deviation is replaced by one fourth of the range. These VI scores (and the intermediate functions $\bar{Q}_j$, $\bar{Q}_{jl}$, and $\bar{Q}^I_{jl}$) can be approximated using the partitioning schemes of Apley and Zhu (2020) as described in Appendix A.2.

Although for notational simplicity we have omitted the dependence of the quantile function on the parameters $\mathscr{W}$, in practice the posterior uncertainty in $\mathscr{W}$ leads to posterior uncertainty in the sensitivity metrics such as $\bar{Q}_j(\tau)$ and $\text{VI}_j(\tau)$. We account for this uncertainty by computing the sensitivity measures for many MCMC samples from the posterior

distribution of $\mathscr{W}$, giving a Monte Carlo approximation of the posterior distribution of the sensitivity measures.

2.6 Simulation

We investigate the numerical performance of our model in four scenarios. The details of each simulation design are provided below.

Design 1. The covariate and response are generated as $X \sim \mathscr{U}(0,5)$ and

$$Y = X + \sin(2X) + 3\epsilon;\ \epsilon \sim \text{Skew-}\mathscr{N}(0,1,4).$$

The quantile curves are parallel, and the data exhibit strong right-skewness.

Design 2. The covariate and response are generated as $X \sim \mathscr{U}(0,1)$ and

$$Y = 3X + [0.5 + 2X + \sin(3\pi X + 1)]\epsilon;\ \epsilon \sim \mathscr{N}(0,1).$$

The data exhibit strong heteroscedasticity. The quantile curves are linear at the median but have strong curvature at the extremes.

Design 3. The covariates $X_j,\ j = 1,2$ are generated from $\mathscr{U}([0,1]\times[0,1])$. The response variable Y is given by

$$Y = \sin(2\pi X_1) + \cos(2\pi X_2) + \sqrt{2(X_1^2 + X_2^2)}\epsilon;\ \epsilon \sim \text{Student's } t(3).$$

The data exhibit both heteroscadasticity and heavy-tailedness.

Design 4. The covariates $X_j,\ j = 1,2,..,d$ are generated from $\mathscr{U}([0,1]^d)$. The quantile func-

tion $Q_Y(\tau|\boldsymbol{X})$ is given by

$$\begin{aligned} Q_Y(\tau|\boldsymbol{X}) = 3(\tau-0.5)&\left(X_1+\frac{3}{5}\right)^3 \\ &+15\left[X_2+4\left(X_2-\frac{1}{2}\right)^2\right]\exp\left(-X_2^2\right) \\ &+12\exp\left[\left(X_3+\frac{1}{2}\right)^2\left(X_4-\frac{1}{2}\right)^2\right] \\ &+5(\tau-1)\left(X_5+\frac{2}{5}\right)\left(X_6+\frac{1}{2}\right)^2+0.25\Phi^{-1}(\tau), \end{aligned}$$

where $\Phi^{-1}(\cdot)$ is the standard normal quantile function and $d \in \{10,20,40\}$. The response variable is generated by sampling $U \sim \mathscr{U}(0,1)$ and setting $Y = Q(U|\boldsymbol{X})$. The quantile process has a complex structure with strong interaction effects. The model is sparse as only the first six covariates affect the quantile function.

For Designs 1–3, we generate samples of sizes $n \in \{50,100,200\}$ and for Design 4 we use $n = 200$. The proposed model is compared to four nonlinear SQR methods: the monotone composite QR neural network (MCQRNN) of Cannon (2018), the nonparametric simultaneous QR (NPSQR) of Das and Ghosal (2018), the nonparametric distribution function simultaneous QR (NPDFSQR) also of Das and Ghosal (2018), and the spectral series conditional density estimator (seriesCDE) of Izbicki and Lee (2016). MCQRNN is implemented in the **qrnn** package in R; codes for NPSQR and NPDFSQR are available from the second author's webpage; and codes for seriesCDE are available from the supplemental material of their online paper. Implementation details including model selection for the competing methods are given in Appendix A.3. For QUINN, we first map the response variable to the unit interval using min-max normalization. The covariates are not required to be normalized. However, it is a common practice to normalize the inputs to a FNN when optimizing its parameters using a gradient-based approach (Bishop et al. 1995). In this chapter, we always map the covariate vector to the unit interval, even if it is one-dimensional. Posterior distribution of QUINN is approximated by 1900 MCMC samples that are obtained by running NUTS for 20,000 iterations, discarding the first 1000 iterations as burn-in and saving every 10th draw from the remaining iterations. Convergence of MCMC is monitored by trace plots of log-likelihood from multiple independent chains as shown in Fig. A.1, and popular diagnostic statistics as described in Appendix A.1.5. The performance of QUINN depends on the number of spline knots p and hidden neurons V, so we use a grid search

approach and select the best combination of $p, V \in \{5, 8, 10\}$ based on WAIC (Watanabe 2013). We choose WAIC over other information criteria (e.g. AIC and DIC) because it is fully Bayesian, uses the entire posterior distribution, and is asymptotically equal to Bayesian leave-one-out cross-validation (Vehtari et al. 2017). Fig. A.2 plots the distribution of out-of-sample RMISE against ranking of WAIC. The result shows that model chosen by WAIC is in favor of a higher out-of-sample prediction accuracy. We also observe that the performance of QUINN is generally robust to different values of the two parameters except for some particularly bad combinations.

To compare the different approaches, 100 data sets are simulated. For each sample size, the performance of each method is measured by the root mean integrated square error (RMISE) between the actual and estimated (posterior mean) quantile processes. We first divide the domain of each dimension of $\boldsymbol{X}$ by g equidistant grid-points, giving $G = g\,d$ vectors $\tilde{\boldsymbol{x}}_1, ..., \tilde{\boldsymbol{x}}_G$ that span the range of $\boldsymbol{X}$. For Designs 1 and 2, we set $g = G = 101$; for Design 3, we set $g = 21$ and thus $G = 21^2$. The RMISE is then approximated as

$$\text{RMISE}(\tau_k) = \sqrt{\frac{1}{G}\sum_{i=1}^{G}\left\{Q_Y(\tau_k|\tilde{\boldsymbol{x}}_i) - \hat{Q}(\tau_k, \tilde{\boldsymbol{x}}_i)\right\}^2}$$

for quantile level $\tau_k \in \{0.05, 0.10, ..., 0.95\}$ and

$$\text{RMISE}_{\text{QP}} = \sqrt{\frac{1}{19}\sum_{k=1}^{19}\text{RMISE}(\tau_k)^2}$$

for the entire quantile process.

The average RMISE_{QP} over 100 simulated data sets along with their standard errors are shown in Table A.1 in Appendix A.4. The results show that QUINN yields significantly smaller average RMISE_{QP} in all settings when compared to NPSQR, NPDFSQR, and seriesCDE; and smaller or similar average RMISE_{QP} in all but one setting when compared to MCQRNN. In particular, QUINN is robust to data sparsity as seen from its small variance. We also plot the average $\text{RMISE}(\tau)$ for cases when $n = 100$ in Fig. 2.1(a)–(c). The results show that QUINN gives the best estimation of intermediate quantiles in all cases, whereas MCQRNN gives better estimation of extreme quantiles when the data exhibit significant heavy-tailedness.

For Design 4, we compare QUINN with MCQRNN and seriesCDE only. We omit NPSQR and NPDFSQR because expanding each covariate of a d-dimensional $\boldsymbol{X}$ using quadratic B-

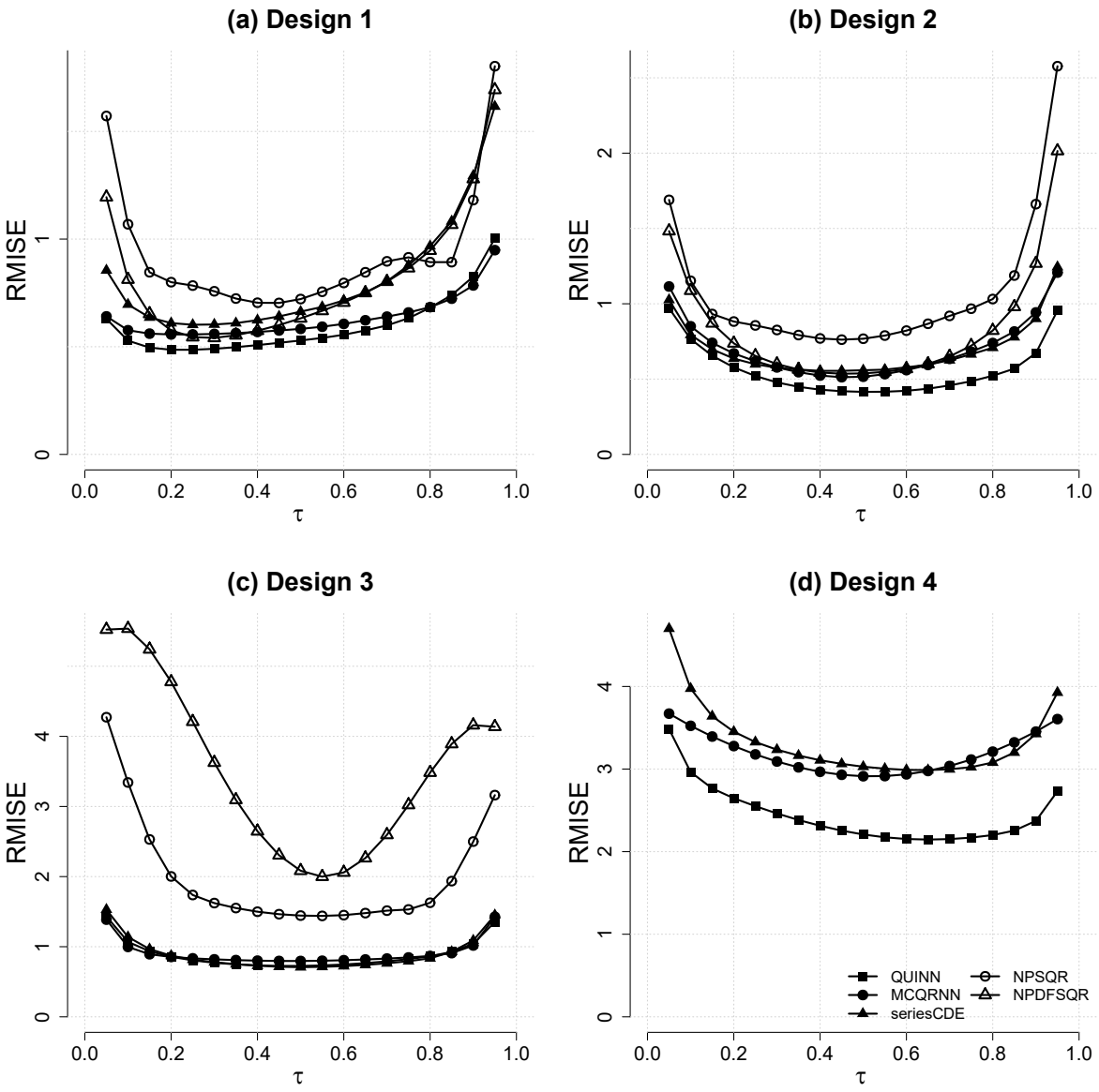


Figure 2.1: RMISE(τ) for the simulation studies at quantile levels $\tau \in \{0.05, 0.1, ..., 0.95\}$. The training sample sizes are $n = 100$ for Designs 1–3 and $n = 200$ for Design 4.

spline basis functions results in a parameter space of dimension $p_{\text{DG}}(p_{\text{DG}}+2)^d$. Even with d as few as 10, fitting NPSQR and NPDFSQR becomes computationally infeasible. We generate sample of size $n = 400$ and split into 200 training and 200 testing data points. Each model is first fit to the training data, then conditional quantile predictions at $\tau \in \{0.05, 0.1, ..., 0.95\}$ are calculated for each given $\boldsymbol{x}$ of the testing data. Posterior distribution of QUINN is

approximated by 2400 samples that are obtained by running NUTS for 50,000 iterations, discarding the first 2000 iterations as burn-in and saving every 20th draw from the remaining iterations. We choose the best model configuration of $p \in \{5, 8, 10\}$ and $V \in \{8, 10, 15\}$ using WAIC. We generate 100 replicates and compare different approaches based on RMISE(τ) and RMISE_{QP} between actual and predicted quantiles conditioned on the testing data points. The average RMISE_{QP} for $d = 10$ are 2.47 for QUINN, 3.21 for MCQRNN, and 3.37 for seriesCDE and the average RMISE(τ) are plotted in Fig. 2.1(d). The result shows that QUINN gives substantially better estimation of the quantile process than MCQRNN and seriesCDE. To further investigate the performance of QUINN in high-dimension setting, we repeat Design 4 with $\boldsymbol{X}$ of dimensions $d = 20, 40$ with the additional covariates being independent of the response. The average RMISE_{QP} are 2.73 and 3.09, respectively. Therefore, QUINN shows promising performance when the quantile process is high-dimensional, has complex interaction effects, and has a sparse structure.

We now demonstrate how ALEs plot can be used with QUINN to visualize main and second-order interaction effects on its predicted quantiles. Design 4 is constructed such that $\bar{Q}_1(\tau, x_1)$, ..., $\bar{Q}_6(\tau, x_6)$, $\bar{Q}_{3,4}(\tau, x_3, x_4)$, and $\bar{Q}_{5,6}(\tau, x_5, x_6)$ are non-zero functions of τ. In addition, $\bar{Q}_j(\tau, x_j)$, $j \in \{1, 5, 6\}$ and $\bar{Q}_{5,6}(\tau, x_5, x_6)$ are non-constant functions of τ. To evaluate the sensitivity of QUINN in identifying these marginal quantile effects, we generate 100 replicates of sample size 5000 from Design 4. For each replicate, we estimate the ALE main effect for each covariate and interaction effect for each pair of covariates at quantile levels $\tau \in \{0.05, 0.10, ..., 0.95\}$ based on the fitted QUINN. The estimated ALE main effects at these quantile levels along with their ground truths are shown in Fig. 2.2, where the thin black lines represent individual estimates based on the 100 simulated data sets, and the thick gray line represents the true ALE effect calculated from the generating model. The estimated ALEs show that QUINN successfully captures the main effects of each covariate. For ALE interaction effects, since it is impossible to visualize all estimated surfaces in one plot, we instead show estimates from 8 randomly selected replicates. For each replicate, estimated $\bar{Q}_{jl}(\tau, x_j, x_l)$ and $\bar{Q}^I_{jl}(\tau, x_j, x_l)$ at quantile levels $\tau \in \{0.05, 0.5, 0.95\}$ are visualized using contour plots (see Appendix A.4) and compared with that of the ground truth. The results show that QUINN also successfully recovers the complex interaction effects of the generating model.

To investigate whether QUINN is capable of recovering the relative importance of the marginal effects, we calculate VI scores for each ALE main and interaction effect. Because only eight marginal effects contribute to the conditional quantile function in Design 4, we

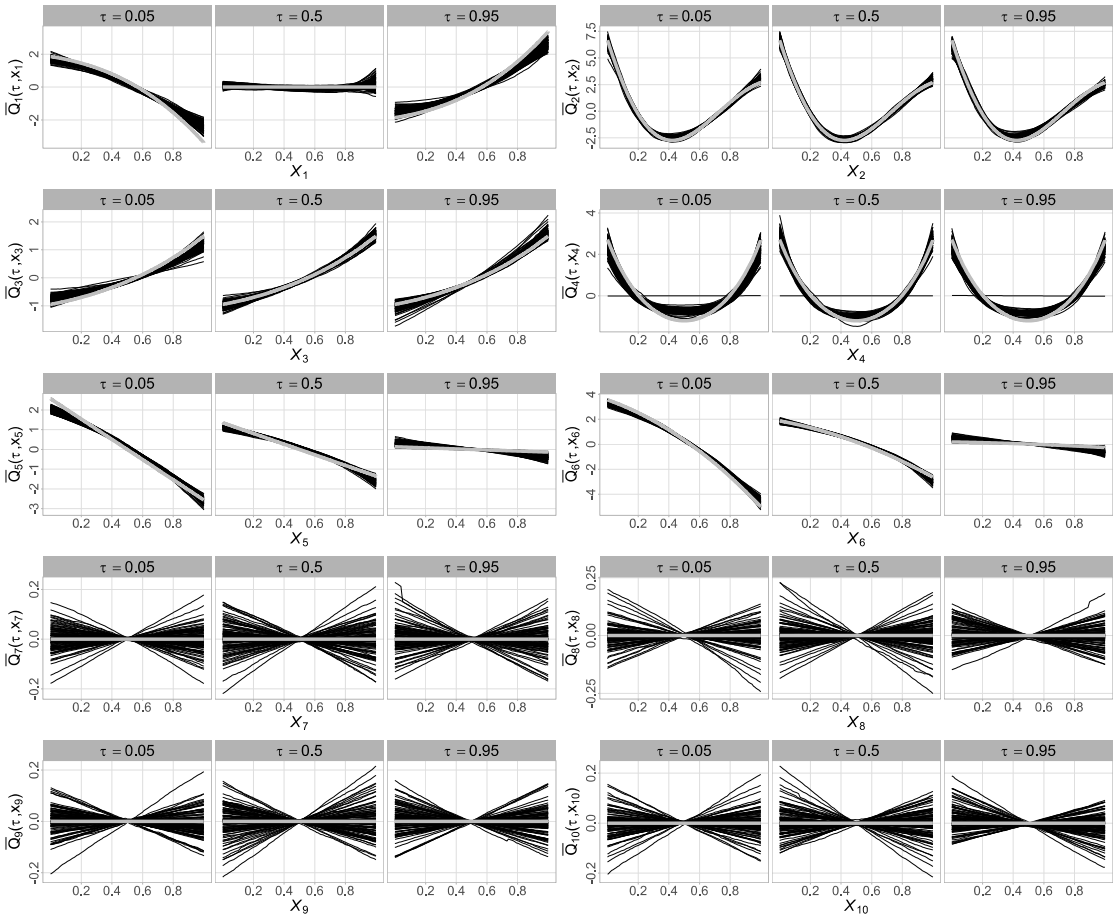


Figure 2.2: Marginal main effect estimates from sensitivity analysis of Simulation Design 4 with $p = 10$. Results above are accumulative local effects (ALE) $\bar{Q}_j(\tau, x_j)$ for each combination of covariate $j \in \{1, ..., 10\}$ and quantile level $\tau \in \{0.05, 0.50, 0.95\}$. Black thin lines represent individual ALE calculated from 100 replicates, and the gray thick line represents the true ALE based on the generating model.

demonstrate the sensitivity of QUINN by showing the rank plot of the top eight estimated marginal effects with the highest VI in Fig. 2.3. The results show that at quantile levels $\tau \in \{0.05, 0.5, 0.95\}$, the estimated VIs and their ranking of the top eight marginal effects resemble the ground truth. The sensitivity analysis shows that QUINN is able to identify the relative order of the important covariate effects.

The simulation results presented in this chapter are only selected ones that demonstrate the predictive accuracy and sensitivity of QUINN. Additional results, including plots of estimated quantile curves and conditional densities, as well as comparison between

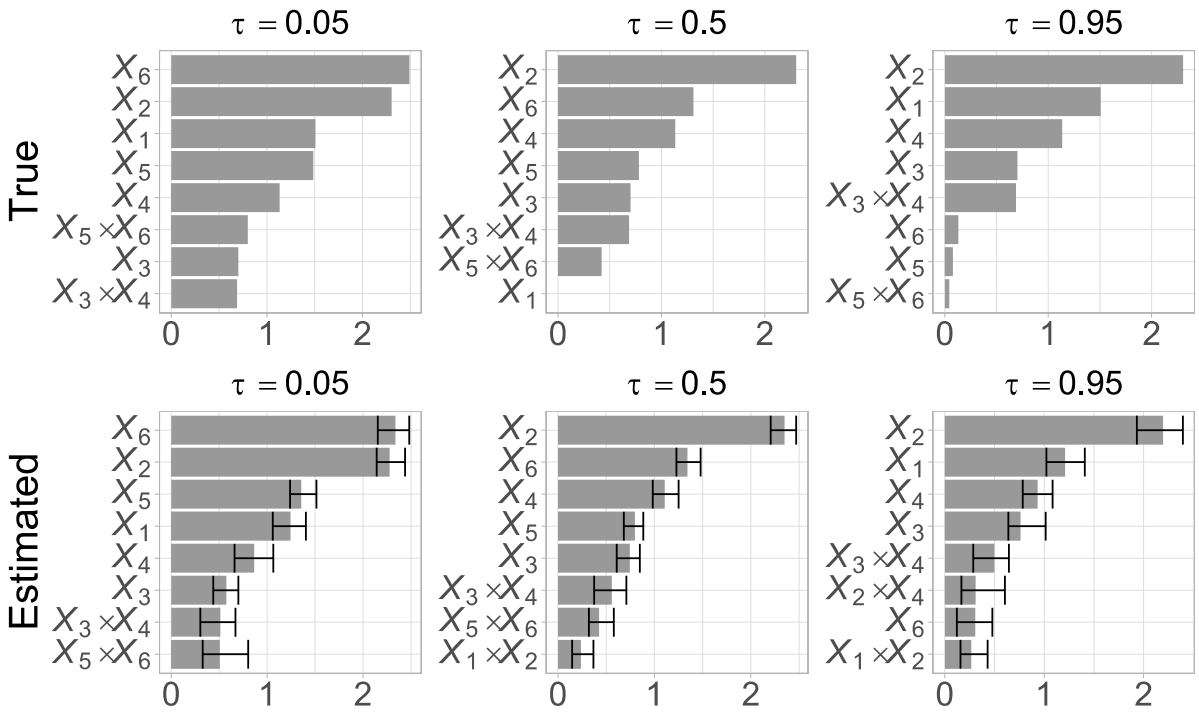


Figure 2.3: Marginal effects importance from sensitivity analysis of Simulation Design 4 with $p = 10$. Results above are estimated and true variable importance of the top 8 marginal effects and quantile levels $\tau \in \{0.05, 0.50, 0.95\}$. Thin horizontal lines represent 95% credible intervals.

estimated and actual ALE second-order effects, are available in Appendix A.4.

2.7 Application to birth weight data

To illustrate the practical effectiveness of QUINN, we study the effect of pregnancy-related factors on infant birth weight (`Weight`, in grams) quantiles. Our data (Xu 2021) consist of 10,000 randomly chosen entries from the 2019 U.S. Natality Data Set (National Center for Health Statistics 2019) on singleton live births to mothers recorded as Black or White, in the age group 18–45, with height between 59 and 73 inches, and smoke no more

than 20 cigarettes daily during pregnancy. The list of covariates contains demographic characteristics, maternal behavior and health characteristics, as well as infant health characteristics. For demographic characteristics, we include indicator of age above 40 years old (`fatherAge`) for the father; and age (`motherAge`, in years), indicators of Black (`Black`), education attainment up to high school graduate (`highSchool`) and at least college graduate (`collegeGraduate`), and parity greater than 1 (`Parity`) for the mother. For maternal behavior and health characteristics, we include body mass index (`BMI`), height (`Height`, in inches), weight gain (`wtGain`, in pounds), indicator of smoking before pregnancy (`Smoker`), average daily number of cigarettes during pregnancy (`Cigaretters`), indicators of not receiving prenatal care (`noPrenatal`), pre-existing diabetes (`preDiab`) and hypertension (`preHype`), gestational diabetes (`gestDiab`) and hypertension (`gestHype`), no infections present and/or treated during pregnancy (`noInfec`), and infertile treatment (`infTreat`). For infant health characteristics, we include gestational age (`Week`) and indicator of boy (`Boy`).

The response variable and all continuous covariates are mapped to the unit interval using min-max normalization. We fit QUINN with $V \in \{10, 20, 30\}$ hidden neurons and $p \in \{10, 15, 20\}$ spline knots. We approximate its posterior distribution using 2400 samples obtained by running NUTS for 50,000 iterations, discarding the first 2000 iterations as burn-in, and selecting every 20th draw from the remaining iterations. The best model configuration is chosen based on WAIC.

To determine which covariates have the most significant impact on the birth weight quantiles, we calculate the ALE-induced VI score for each covariate across different quantile levels. In particular, we are interested in identifying the covariates that most impact LBW (represented by the 0.05 quantile), typical birth weight (TBW, represented by the 0.5 quantile), and HBW (represented by the 0.95 quantile). Fig. 2.4 shows the ranking of ALE main effects at $\tau \in \{0.05, 0.50, 0.95\}$. The main effects of `Week`, `height`, `BMI`, `wtGain`, `Cigarette`, `Black`, `preDiab`, `Boy` and `Smoker` have the highest VI measure at all three quantiles and therefore are most influential on the birth weight distribution. In particular, `Week` has a dominant effect on all three quantiles, `Cigarette` is most influential on LBW, and `Height` and `BMI` are more influential on TBW and HBW.

To understand the functional relationship between the top covariates and the predicted birth weight quantiles of QUINN, we plot their estimated ALE main effects $\hat{\tilde{Q}}_j(\tau)$ at $\tau \in \{0.05, 0.50, 0.95\}$ in Fig. 2.5(a)–(i). The results show that higher values of `Week`, `height`, `BMI`, `wtGain`, `preDiab` and `Boy` are associated with higher predicted birth weight, whereas

higher values of Cigarette, Black and Smoker are associated with lower predicted birth weight. Furthermore, the effects of Week, height, BMI, wtGain, Cigarette and preDiab on birth weight are significantly non-constant across quantiles, and the effects of Week, BMI, wtGain, Cigarette are highly nonlinear. For example, as Cigarette increases, LBW and TBW display a consistent downward trend, whereas HBW plateaus when Cigarette is greater than 13; as wtGain increases, HBW and TBW display a consistent upward trend, whereas LBW plateaus when wtGain is greater than 65.

The results in Fig. 2.5(a)–(i) also provide numerical quantification of the main effects on the predicted birth weight quantiles. For example, Fig. 2.5(b) shows that compared to mothers who do not smoke during pregnancy, mothers who smoke as many as 20 cigarettes daily are associated with a more than 250-gram decrease in predicted LBW, on average. Fig. 2.5(g) shows that compared to mothers who do not have pre-existing diabetes, mothers who have pre-existing diabetes are associated with a 254-gram increase in predicted HBW, on average.

In addition to covariate effects on specific quantiles, QUINN also allows direct characterization of covariate effects on the whole conditional distribution thanks to its density regression nature. To illustrate this property of QUINN, Fig. 2.5(j) plots the predicted birth weight density for Week $\in \{33, 34, ..., 42\}$. The result shows that gestational age has a prominent effect on the location of the predicted density. The shifting of the density is most significant when gestational age increases from 33 to 37 and gradually plateaus when the pregnancy term further increases. For each predicted density, we also highlight the region of LBW (Weight<2500) and HBW (Weight>4000). The result indicates that preterm (Week<37) and postterm (Week>40) are determinant factors of LBW and HBW, repsectively. Fig. 2.5(k) plots the predicted birth weight CDF for different levels of preDiab. Compared to mothers who do not have pre-existing diabetes, mothers with pre-existing diabetes are associated with a 10% increase in probability of giving birth to an overweight infant.

We further analyze the second-order interaction effects between the top covariates that have significant main effects. For each combination, we estimate its ALE joint effect $\bar{Q}_{jl}(\tau)$ as well as its pure interaction effect $\bar{Q}^I_{j,l}(\tau)$ at $\tau \in \{0.05, 0.5, 0.95\}$. The significance of each interaction is determined by the estimated VI measure $\widehat{\text{VI}}_{jl}(\tau)$. Among all interaction considered, the interaction between gestational age and average daily number of cigarettes during pregnancy (Week × Cigarette) yields the highest VI measure on all three quantiles. To understand the functional relationship between Week × Cigarette and the birth weight quantiles, Fig. 2.6 plots the contour of the joint effects of Week and Cigarette, as quantified

by $\bar{Q}_{jl}(\tau)$. The result indicates that higher gestational age is associated with higher birth weight irregardless of maternal smoking habit, but the effect is clearly amplified for non-smokers. In addition, heavier maternal smoking is associated with lower birth weight only for births that occur after the 37th week of pregnancy.

2.8 Discussion

In this chapter, we propose a novel nonlinear SQR model that leverges the flexibility of spline and neural network. We adopt a Bayesian framework by assigning prior distributions to the weight parameters and utilize the state-of-art NUTS to sample efficiently from the high-dimensional posterior. Compared to existing works, our method models the full quantile process, does not involve constrained optimization, and scales to high-dimensional setting. We also show that our model can yield meaningful interpretation via ALE plots and variable importance scores. Simulation studies show that our model better recover high-dimensional quantile process with complex structure and is robust to data and model sparsity. Sensitivity analysis shows that our can accurately captures quantile-dependent covariate effects.

The proposed method was used to analyze the relationship between birth weight and pregnancy-related factors of U.S. newborns and specifically to identify influential effects on LBW and HBW. Our results showed that LBW is primarily associated with prematurity and heavy maternal smoking; whereas HBW is primarily associated high maternal body mass index, maternal height, maternal weight gain, and having pre-existing diabetes. Future extension can focus on accommodating spatial and/or temporal correlation between observations and variable/model selection using sparsity-inducing priors.

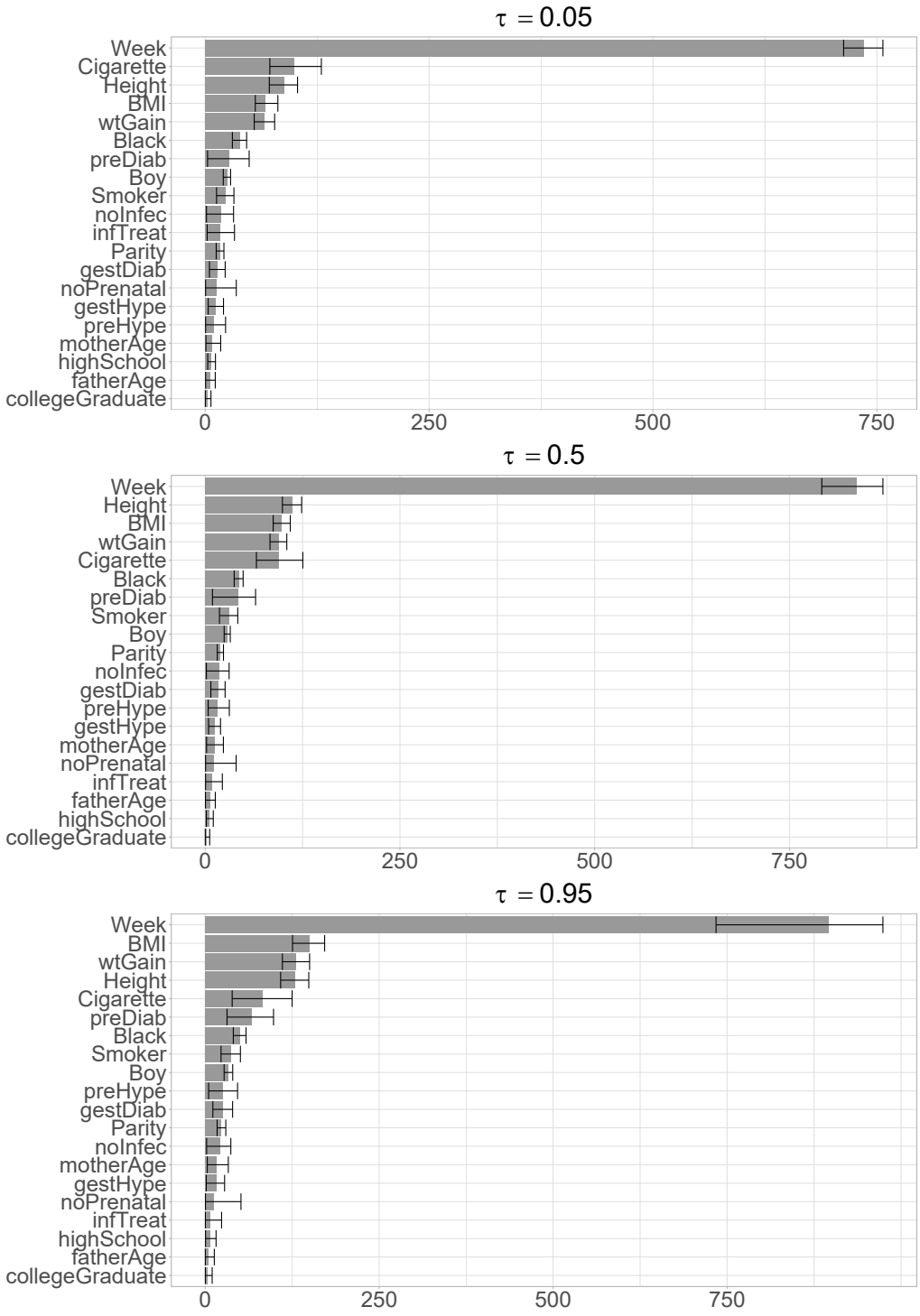


Figure 2.4: Variable importance of the birth weight data. Results above are posterior mean variable importance measure of all main effects at quantile levels $\tau \in \{0.05, 0.50, 0.95\}$. The thin horizontal lines represent 95% credible intervals.

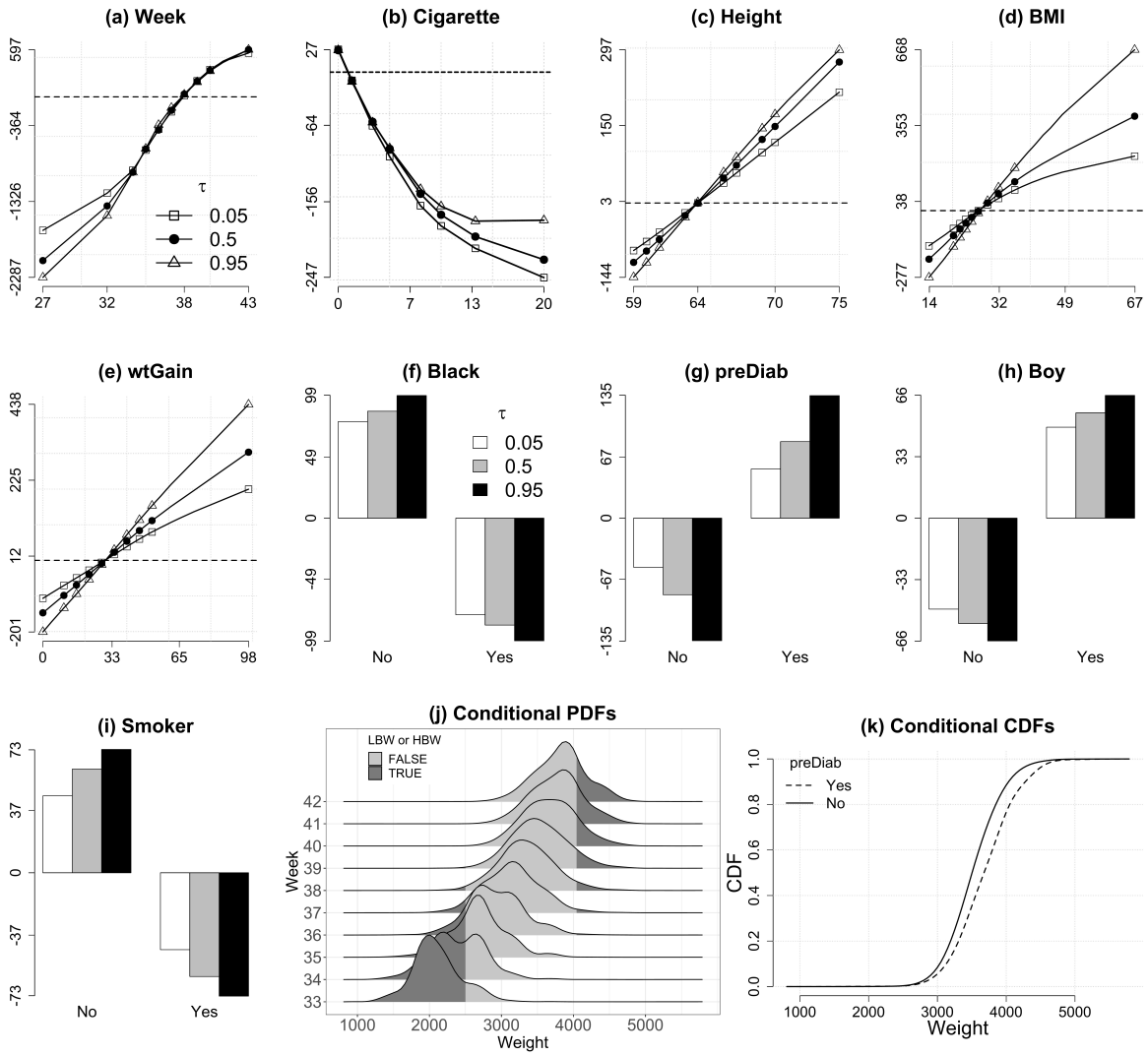


Figure 2.5: Marginal main effect estimates of the birth weight data. (a)–(i) Posterior mean ALE main effects at quantile levels $\tau \in \{0.05, 0.50, 0.95\}$ for the top 9 important covariates. For continuous covariates, black dashed line represents the value 0. (j)-(i) Conditional distribution estimates by gestational age (`Week`) and pre-pregnancy diabetes indicator (`preDiab`), respectively, with all other covariates fixed at their median (continuous covariates) or mode (binary covariates).

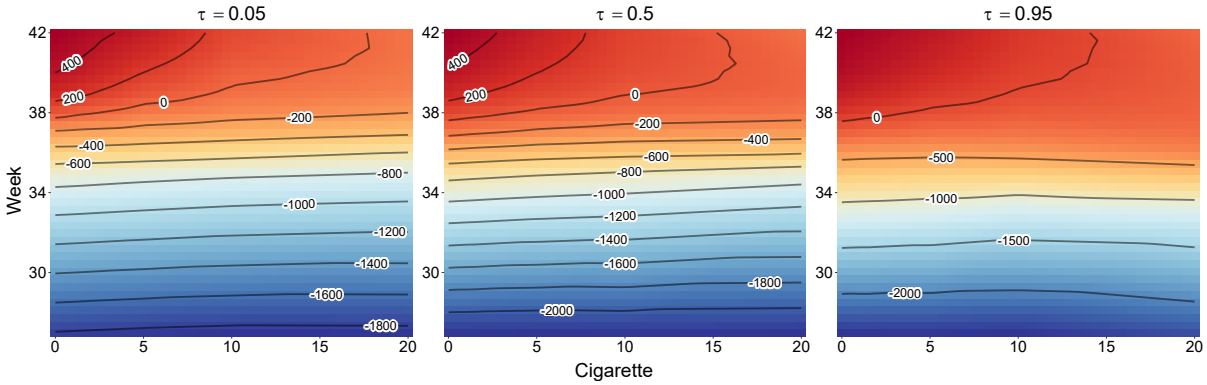


Figure 2.6: Marginal joint effect estimates of the birth weight data. Results above are posterior mean ALE joint effects of gestational age (`Week`) and average daily number of cigarettes (`Cigarette`) at quantile levels $\tau \in \{0.05, 0.5, 0.95\}$. Regions where estimated quantile value is high are colored in red, and regions where estimated quantile level is low are colored in blue. This figure appears in color in the electronic version of this article, and any mention of color refers to that version.

CHAPTER

3

AN R PACKAGE FOR SEMIPARAMETRIC QUANTILE REGRESSION

The work in this Chapter has been submitted as a paper entitled “SPQR: An R Package for Semiparametric Density and Quantile Regression".

3.1 Background

Most existing implementations of quantile regression (QR) in R assume a linear relationship between the conditional quantile and covariates, examples include **bayesQR** (Benoit and Van den Poel 2017), **lqr** (Galarza et al. 2022) and **quantreg** (Koenker 2022). These methods become too restrictive, leading to large estimation bias, when the underlying relationship is highly nonlinear. To overcome this challenge, semiparametric QR models that allow some or some transformation of the covariate effects be modeled flexibly have been proposed, examples are **quantreg.nonpar** (Lipsitz et al. 2016), **plaqr** (Maidman 2017) and **qgam** (Fasiolo et al. 2021) that use generalized additive models; and **siqr** (Yu 2021) that uses a single

index model. Additive models allow flexible and interpretable modeling of main effects, but explicitly specifying the interaction terms quickly become a tedious task as the data dimension increases. Single index model alleviates curse of dimensionality by projecting the covariates to a 1-dimensional space, but when the intrinsic data dimension is high such projection can be too coarse to retain all valuable information. A more serious disadvantage shared by all aforementioned methods is that they do not ensure the monotonicity of the estimated conditional quantile function. Without such constraint, the estimated quantiles can cross each other when the sample size is small, leading to a invalid response distribution that is difficult or impossible to draw inference from.

Few existing packages are capable of estimating multiple quantiles under non-crossing constraints. The **qrjoint** package (Tokdar and Cunningham 2019) directly models the linear coefficient function as a monotone process and allows simultaneous estimation of non-crossing quantile planes. The `mcqrnn()` function in the **qrnn** package (Cannon 2018) can model non-linear non-crossing quantile curves by treating the quantile level as an observed covariate and imposing positivity constraints on its correspond weights in the neural network. However, the model suffers from high variance when the sample size is small (Xu and Reich 2021) and uncertainty quantification is not straightforward.

3.2 Contribution

The purpose of this chapter is to introduce a new R package, named **SPQR** (Xu and Majumder 2022), for flexible QR modeling. This package implements the semiparametric QR (SPQR) model proposed by Xu and Reich (2021) which allows simultaneous estimation of non-linear non-crossing quantile curves. The method begins by specifying a semiparametric model for the conditional distribution function using shape-constrained splines, the coefficients are then modeled nonparametrically as functions of covariates using artificial neural networks. As a result, valid estimates of the conditional response distribution and its quantiles can be simultaneously obtained. **SPQR** provides three approaches for fitting SPQR: maximum likelihood estimation (MLE) and maximum *a posteriori* probability (MAP) which provide point estimates, and Markov chain Monte Carlo (MCMC) which provides uncertainty quantification through posterior samples. To our best knowledge and at the time of writing, **SPQR** is the only package that implements a fully Bayesian framework for estimating semiparametric non-crossing QR. The main computations of **SPQR** rely on **torch**

(Falbel and Luraschi 2022) and **Rcpp** (Eddelbuettel et al. 2022a). Specifically, MLE and MAP are optimized using the Adam routine (Kingma and Ba 2014) in **torch** and take advantage of its GPU capability, and MCMC is implemented using **Rcpp** and **RcppArmadillo** (Eddelbuettel et al. 2022b) for efficient gradient computation. To increase the interpretability and transparency of SPQR, **SPQR** also provides function to compute quantile accumulative local effects (ALE; Apley and Zhu 2020) to characterize and visualize quantile-dependent main and interaction effects, as well as function to compute quantile-dependent variable importance measures.

3.3 Organization

The rest of the chapter is organized as follows. We first give a brief review of the methodology background of SPQR and quantile ALEs in Secton 3.4. We then introduce the implemented computation approaches for fitting SPQR in Section 3.5. Section 3.6 introduces the **SPQR** package and its main features in detail. In Section 3.7, we demonstrate the usage and effectiveness of **SPQR** in density and QR problems using simulated and real data. In particular, we apply our package to the Australia electric demand data, available in the **qgam** package, and analyze the quantile effects of temperature and time on residential electric energy consumption. Section 3.8 concludes the chapter with some discussion.

3.4 Methodology

3.4.1 Density regression model

To estimate quantile effects, we first build a model for the probability density function (PDF) of the response Y conditioned on covariates $\boldsymbol{X} = (X_1, ..., X_p)$. We assume that the response and all p covariates are scaled to $[0,1]$ to simplify the specifications of prior distributions and basis functions. The PDF is assumed to be a function of K second-order M-spline basis functions, $\{M_k(y) : 1 \leq k \leq K\}$, with equally-spaced knots spanning $[0,1]$. Each M-spline basis function is itself a valid PDF on $[0,1]$ (Ramsay 1988), and therefore any convex combination of these functions is also a valid PDF. The model is

$$f(y|\boldsymbol{X}) = \sum_{k=1}^{K} \theta_k(\boldsymbol{X}) M_k(y),$$

where the probabilities $\theta_k(\boldsymbol{X})$ satisfy $\theta_k(\boldsymbol{X}) \geq 0$ and $\sum_{k=1}^{K} \theta_k(\boldsymbol{X}) = 1$ for all possible $\boldsymbol{X}$.

To provide flexibility and ensure non-negative weights $\theta_k(\boldsymbol{X})$ that sum to one, we use a fully connected neural network (NN) with softmax output activation. An NN with L layers ($L-1$ hidden layers) is parameterized by a set of weight matrices $\mathscr{W} = \{\boldsymbol{W}^{(1)}, ..., \boldsymbol{W}^{(L)}\}$, with $\boldsymbol{W}^{(l)} \in \mathbb{R}^{(V_l+1)\times V_{l-1}}$ where V_l is the number of units (excluding the intercept/bias node) in layer l. We define the input layer as layer 0 for simplicity. The first layer is

$$z_i^{(1)}(\boldsymbol{X}, \mathscr{W}) = W_{i0}^{(1)} + \sum_{j=1}^{p} W_{ij}^{(1)} X_j.$$

We make the functional dependence on $\boldsymbol{X}$ and $\mathscr{W}$ explicit in our notation as it will help clarify what follows. Subsequent hidden layers are defined by the recursion

$$\begin{aligned} u_i^{(l)}(\boldsymbol{X}, \mathscr{W}) &= \phi\left\{z_i^{(l)}(\boldsymbol{X}, \mathscr{W})\right\} \quad \text{for } \; l \in \{1, ..., L-1\} \\ z_i^{(l+1)}(\boldsymbol{X}, \mathscr{W}) &= W_{i0}^{(l+1)} + \sum_{j=1}^{V_l} W_{ij}^{(l+1)} u_j^{(l)}(\boldsymbol{X}, \mathscr{W}) \end{aligned} \tag{3.1}$$

where ϕ is the activation function taken to be either the hyperbolic tangent function $\phi(u) = (e^{2u}-1)/(e^{2u}+1)$ or the rectified-linear function $\phi(u) = \max(0, u)$. Finally, the softmax activation is used in the output layer to ensure probabilities sum to one

$$\theta_k(\boldsymbol{X}, \mathscr{W}) = \frac{\exp\left\{z_k^{(L)}(\boldsymbol{X}, \mathscr{W})\right\}}{\sum_{i=1}^{K} \exp\left\{z_i^{(L)}(\boldsymbol{X}, \mathscr{W})\right\}}.$$

The SPQR model for the conditional PDF is then

$$f(y|\mathscr{W}, \boldsymbol{X}) = \sum_{k=1}^{K} M_k(y) \frac{\exp\left\{z_k^{(L)}(\boldsymbol{X}, \mathscr{W})\right\}}{\sum_{i=1}^{K} \exp\left\{z_i^{(L)}(\boldsymbol{X}, \mathscr{W})\right\}}. \tag{3.2}$$

Endowed with the approximation theories of spline and NN, the model in (3.2) can approximate any smooth PDF as K and the V_l increase. Therefore, this model can capture complex relationships, such as covariate-dependent variance, skewness, or likelihood of extreme events.

Computing the PDF is simple and fast given the parameters $\mathscr{W}$, and since the integral of M-spline functions are I-spline functions (Ramsay 1988), (3.2) immediately gives rise to

an expression for the cumulative distribution function (CDF)

$$F(y|\mathscr{W}, \boldsymbol{X}) = \sum_{k=1}^{K} I_k(y) \frac{\exp\left\{z_k^{(L)}(\boldsymbol{X}, \mathscr{W})\right\}}{\sum_{i=1}^{K} \exp\left\{z_i^{(L)}(\boldsymbol{X}, \mathscr{W})\right\}}, \tag{3.3}$$

where $I_k(y)$ are I-spline basis functions. The conditional quantile function for quantile level $\tau \in (0,1)$ is defined as the function $Q(\tau|\mathscr{W}, \boldsymbol{X})$ so that

$$F\{Q(\tau|\mathscr{W}, \boldsymbol{X})|\mathscr{W}, \boldsymbol{X}\} = \tau.$$

The conditional quantile function for this model is not available in closed-form, but can be approximated by numerically inverting $F(y|\mathscr{W}, \boldsymbol{X})$. Given that (3.3) models a valid CDF, the conditional quantile function estimated through this approach satisfies the non-crossing constraint

$$\frac{\partial Q(\tau|\mathscr{W}, \boldsymbol{X})}{\partial \tau} > 0, \ \forall \ \boldsymbol{X}$$

and does not require any second-stage monotonization treatment.

3.4.2 Summarizing covariate effects on quantiles

SPQR has the advantage of being a flexible semiparametric model that can capture complex non-linear covariate effects on various aspects of the response distribution. A disadvantage is that it is difficult to interpret individual parameters because the weights $\mathscr{W}$ are not individually identified and do not correspond to meaningful quantities. In most applications where QR is used, understanding the covariate effect on different quantiles is of paramount interest. Therefore, we seek to quantify covariate effects on specific aspects of the response distribution as measured by the quantile function $Q(\tau|\boldsymbol{X})$.

We quantify covariate quantile effects using the accumulative local effects (ALEs) of Apley and Zhu (2020). The sensitivity of $Q(\tau|\mathscr{W}, \boldsymbol{X})$ to covariate j is naturally quantified by the partial derivative

$$q_j(\tau|\mathscr{W}, \boldsymbol{X}) = \frac{\partial Q(\tau|\mathscr{W}, \boldsymbol{X})}{\partial X_j}.$$

The ALE begins by averaging $q_j(\tau|\mathscr{W}, \boldsymbol{X})$ over $\boldsymbol{X}$ conditioned on $X_j = u$,

$$\bar{q}_j(\tau|\mathscr{W}, u) = \mathbb{E}_{\boldsymbol{X}}\{q_j(\tau|\mathscr{W}, \boldsymbol{X})|X_j = u\}.$$

The ALE main effect function of X_j is then defined as

$$ALE_j(\tau|\mathscr{W}, x) = \int_0^x \bar{q}_j(\tau|\mathscr{W}, u)du.$$

Analogous formulas define the second-order ALE interaction effect for X_j and X_l, $\text{ALE}_{jl}(\tau|\mathscr{W}, x_j, x_l)$, by taking the partial derivative with respect to both X_j and X_l. These functions can be plotted by τ to summarize how the predicted quantile changes with respect to change in the covariate values. In addition to the ALEs, we follow Greenwell et al. (2018) and distill the ALE function to one-number summaries to compare variable importance by quantile level. The variable importance (VI) for continuous covariates are characterized by the standard deviation of the ALE with respect to the marginal distribution of $\boldsymbol{X}$, i.e., $\text{VI}_j(\tau|\mathscr{W}) = \text{SD}\{ALE_j(\tau|\mathscr{W}, X_j)\}$ and $\text{VI}_{jl}(\tau|\mathscr{W}) = \text{SD}\{ALE_{jl}(\tau|\mathscr{W}, X_j, X_l)\}$. For discrete covariates with few unique levels, the standard deviation is replaced by the range.

The ALE and VI summaries depend on the model parameters, $\mathscr{W}$. They can either be evaluated using a point-estimate $\widehat{\mathscr{W}}$ to give a point-estimate of the summaries, $ALE_j(\tau|\widehat{\mathscr{W}}, x)$ and $\text{VI}_j(\tau|\widehat{\mathscr{W}})$, or, for a Bayesian analysis, the posterior samples can be used to quantify uncertainty of the summaries such as the posterior probability that variable j is more important than variable l for predicting conditional quantile at τ.

3.5 Computational approaches

SPQR includes four computational algorithms to estimate the model parameters and importance measures: maximum likelihood estimation (MLE), maximum *a posteriori* probability (MAP), Hamiltonian Monte Carlo (HMC) and the no-U-turn sampler (NUTS). These algorithms are described in detail below.

3.5.1 Maximum likelihood estimation (MLE)

Directly modeling the conditional PDF means that the negative likelihood function given n training observations $(\boldsymbol{X}_i, y_i)$ for $i \in \{1, ..., n\}$ has a closed form,

$$\ell(\mathscr{W}) = -\sum_{i=1}^{n} \log\left\{\sum_{k=1}^{K} M_k(y_i)\theta_k(\boldsymbol{X}_i, \mathscr{W})\right\} \tag{3.4}$$

and can be used as a loss function to be minimized, leading to the MLE estimator

$$\widehat{\mathscr{W}}_{\text{MLE}} = \arg\min_{\mathscr{W}} \ell(\mathscr{W}). \tag{3.5}$$

Equation (3.5) can be solved using standard back-propagation algorithms. The MLE estimator, however, does not put any constraint on the magnitude of $\mathscr{W}$ and therefore can result in an unstable model that overfits the data. One solution is to augment the loss function in (3.4) with a regularization term that penalizes large $\mathscr{W}$, for example the *weight decay* regularization. However, carefully choosing the penalty coefficients is crucial to the predictive performance but increases computation complexity. Furthermore, uncertainty analysis of the estimators obtained by (3.5) is not straightforward and has to rely on bootstrap approaches, bringing additional computational challenges to the already complex problem.

3.5.2 Bayesian estimation

To address the lack of uncertainty quantification posed by MLE, we adopt a Bayesian framework and give prior distribution for the weight parameters. The uncertainty of the SPQR estimators is then characterized by their posterior distributions. Specifying prior distributions also provides regularization to the model and stabilizes weight estimation. We assume hierarchical normal priors for the weights

$$W_{ij}^{(l)} | \sigma^{(l)}, \lambda_j^{(l)} \sim \mathcal{N}\left(0, \sigma^{(l)2} \lambda_j^{(l)2}\right),\ p(\lambda_j^{(l)}) \sim p(\lambda_j^{(l)}; \gamma_\lambda) \ \text{ for } \ j \geq 0 \tag{3.6}$$

where $\sigma^{(l)}$ is the layer-wise global scale shared by all weights in layer l, which can either be set to a constant value or estimated using non-informative priors, and $\lambda_j^{(l)}$ is a unit-wise local scale with hyper-prior $p(\lambda_j^{(l)}; \gamma_\lambda)$. Hierarchical normal distribution is the most commonly used prior for NN weights as they impose a *weight decay* penalty with estimable penalty coefficients, and many priors proposed in the Bayesian NN literature are variants of (3.6). In **SPQR**, we consider three of such models as summarized in Table 3.1: the Gaussian Process (GP) prior, the Automatic Relevance Determinantion (ARD) prior, and the Gaussian Scale Mixture (GSM) prior.

The GP prior is proposed by Neal (1996) who considers the weights and bias in each layer as separate parameter blocks. The weights in each layer depend on a common variance, $\lambda^{(l)}$,

and the bias is given a separate variance. In addition, the variances on weights are scaled by the width of the layer, V_{l-1}, for all but the input layer. The NN under such setting is shown to converge to a certain multi-output Gaussian process under the condition that $V_l \to \infty$ for $l \in [1, L-1]$ in both the case of $L = 2$ (Neal 1996) and $L \to \infty$ (Matthews et al. 2018). The GP prior assumes that the weights marginally follow Gaussian distributions, since they share a common variance hyperparameter. Recent studies on Bayesian NNs, however, have found that distribution of weights in a deep NN (DNN) can be heavy-tailed (Vladimirova et al. 2019; Fortuin et al. 2021). Therefore, it might be helpful to incorporate such knowledge and allow a wider model coverage for the prior distributions for weights. The ARD prior was originally developed by MacKay (1992) who assigns weights in the input-to-hidden layer unit-wise local scales so that the magnitude of weights associated with each input will determine its relevance. Under this setting, the marginal distribution of the input weights, $W_{ij}^{(1)}|\sigma^{(1)}$, depends on the hyper-prior for local scales, $\lambda_j^{(1)}$, through the integration

$$p(w_{ij}^{(1)}|\sigma^{(1)}) = \int \mathcal{N}(0, \sigma^{(1)2}\lambda_j^{(1)2}) p(\lambda_j^{(1)}; \gamma_\lambda) d\lambda_j^{(1)}. \tag{3.7}$$

For all other layers, the ARD prior has the same structure as that of the GP prior. The GSM prior is a direct generalization of the ARD prior by allowing the layer-wise global scale $\sigma^{(l)}$ to be estimable and all layers to have the flexibility of (3.7). Therefore, it not only determines the relevance of each input feature but also that of each latent feature in deeper layers.

To complete the prior specification, we assign the local scale, $\lambda_j^{(l)}$, non-informative hyper-priors. In the case of GSM, we also specify a non-informative hyper-prior for the global scale, $\sigma^{(l)}$. A common choice is the inverse-Gamma distribution,

$$\begin{aligned} p(\cdot; \gamma_\lambda) &= \textit{Inv-}\mathcal{G}\textit{amma}(a_\lambda, b_\lambda) \\ p(\cdot; \gamma_\sigma) &= \textit{Inv-}\mathcal{G}\textit{amma}(a_\sigma, b_\sigma), \end{aligned} \tag{3.8}$$

which is also what will be assumed in this chapter. Under this setting, the marginal distribution of weights $p(w_{ij}^{(l)}|\sigma^{(l)})$ follows a Student-t whose degrees-of-freedom is determined by the hyperparameters, a_λ and b_λ. Notice that in the cases of GP prior, by the scaling property of Gamma distributions, the prior for $\lambda_j^{(l)}$, $l \geq 2$ is also inverse-Gamma

$$\lambda_j^{(l)} \overset{indep}{\sim} \textit{Inv-}\mathcal{G}\textit{amma}(a_\sigma, \frac{b_\sigma}{V_{l-1}}) \quad \text{for} \quad j \geq 2$$

Table 3.1: Prior distributions: **SPQR** allows for several models for the prior distribution for the layer-wise global scale, $\sigma^{(l)}$, and unit-wise local scale, $\lambda_j^{(l)}$. This table gives the Gaussian Process (GP), Automatic Relevance Determination (ARD) and Gaussian Scale Mixture (GSM) priors in terms of unit index in layer l, $j \in [0, V_l]$, and hyperparameters γ_σ and γ_λ.

Name of the prior	$\sigma^{(l)}$	$\lambda_0^{(l)}$	$\lambda_j^{(l)},\ \forall j \geq 1$
GP	1	$p(\lambda_0^{(l)};\gamma_\lambda)$	$\lambda^{(l)} \sim p(\lambda^{(l)};\gamma_\lambda)$ $\lambda_j^{(1)} = \lambda^{(1)};\ \lambda_j^{(l)} = \lambda^{(l)}/V_{l-1},\ \forall l \geq 2$
ARD	1	$p(\lambda_0^{(l)};\gamma_\lambda)$	$\lambda_j^{(1)} \overset{iid}{\sim} p(\lambda_j^{(1)};\gamma_\lambda)$ $\lambda_j^{(l)} \sim \text{GP},\ \forall l \geq 2$
GSM	$p(\sigma^{(l)};\gamma_\sigma)$	$p(\lambda_0^{(l)};\gamma_\lambda)$	$\lambda_j^{(l)} \overset{iid}{\sim} p(\lambda_j^{(l)};\gamma_\lambda)$

Maximum *a posteriori* estimation (MAP)

Before we introduce the fully Bayesian approach to estimate NN parameters using HMC, we note that the prior specification in (3.7) enables the adoption of a MAP method to estimate the weights, i.e.,

$$\begin{aligned}\widehat{\mathscr{W}}_{\text{MAP}},\ \hat{\boldsymbol{\sigma}},\ \hat{\boldsymbol{\Lambda}} &= \underset{\mathscr{W},\boldsymbol{\sigma},\boldsymbol{\Lambda}}{\arg\min}\left\{\ell(\mathscr{W}) - \log p(\mathscr{W},\boldsymbol{\sigma},\boldsymbol{\Lambda})\right\} \\ p(\mathscr{W},\boldsymbol{\sigma},\boldsymbol{\Lambda}) &= \prod_{k=1}^{K}\prod_{l=1}^{L}\prod_{j=0}^{V_l}\mathcal{N}(W_{kj}^{(l)}|0,\sigma^{(l)}\lambda_j^{(l)})p(\sigma^{(l)}|\gamma_\sigma)p(\lambda_j^{(l)}|\gamma_\lambda).\end{aligned} \tag{3.9}$$

Solving the optimization problem in (3.9) will not lead to sensible results as the gradients of $\sigma^{(l)}$ and $\lambda_j^{(l)}$ do not depend on the data. We adopt a reparameterization of (3.9) to let the likelihood function become direct function of the scale hyperparameters

$$\begin{aligned}\widehat{\mathscr{Z}},\ \hat{\boldsymbol{\sigma}},\ \hat{\boldsymbol{\Lambda}} &= \underset{\mathscr{Z},\boldsymbol{\sigma},\boldsymbol{\Lambda}}{\arg\min}\left\{\ell(\mathscr{W}) - \log p(\mathscr{Z},\boldsymbol{\sigma},\boldsymbol{\Lambda})\right\} \\ W_{kj}^{(l)} &= \sigma^{(l)}\lambda_j^{(l)}Z_{kj}^{(l)} \\ p(\mathscr{Z},\boldsymbol{\sigma},\boldsymbol{\Lambda}) &= \prod_{k=1}^{K}\prod_{l=1}^{L}\prod_{j=0}^{V_l}\mathcal{N}(Z_{kj}^{(l)}|0,1)p(\sigma^{(l)}|\gamma_\sigma)p(\lambda_j^{(l)}|\gamma_\lambda).\end{aligned} \tag{3.10}$$

Equation (3.10) can be solved by substituting the prior distributions for variance hyperparameters with any of the three models described in Table 3.1. The MAP estimates of $\mathscr{W}$ can then be calculated as $\widehat{W}^{(l)}_{kj,\text{MAP}} = \hat{\sigma}^{(l)}\lambda^{(l)}_j \hat{Z}^{(l)}_{kj}$. As with the MLE method, MAP estimation uses standard back-propagation algorithms, provides only a point estimate of $\mathscr{W}$ and does not allow for direct quantification of model uncertainty.

Hamiltonian Monte Carlo (HMC)

MCMC produces samples from the posterior distribution of the parameters that can be used to approximate their entire posterior distribution. We follow the strategy of Neal (1996) and use a block-updating scheme that utilizes two MCMC algorithms. The conditional distribution of weights given the variance hyperparameters and the data are approximated using the Hamiltonian Monte Carlo (HMC) sampler, whereas the conditional distribution of variance hyperparameters given the weights are approximated using the Gibbs sampler (Geman and Geman 1984). HMC permits efficient sampling from a high-dimensional target distribution by using its gradient with respect to each direction. It uses an approximate Hamiltonian dynamics simulation based on numerical integration which allows the sampler to explore more carefully in regions with high density and escape quickly from regions with low density (Betancourt 2017). The candidate value of $\mathscr{W}$ found by this simulation is then accepted with a Metropolis step to correct for any numerical error resulted from the numerical integration.

We consider two implementations of the HMC sampler: the vanilla HMC sampler (Neal 2011) and the more advanced no-U-turn sampling (NUTS; Hoffman and Gelman 2014). The vanilla HMC sampler requires setting the number of approximate integration time t, which is the number of leap-frog steps L_ϵ multiplied by the step size ϵ. The step size is automatically optimized during warmup sample iterations using dual-averaging to match an acceptance-rate target, and the number of steps will be calculated as $L_\epsilon = \lfloor t/\epsilon \rfloor$. An Euclidean metric $\mathbf{M}$, where $\mathbf{M}^{-1}$ estimates the posterior covariance of $\mathscr{W}$, is also estimated during warmup to help project the parameters to a space where sampling can be done more efficiently. The NUTS, compared to HMC, has the further advantage of adaptively setting the number of leap-frog steps L_ϵ on the fly during both warmup and sampling. This greatly reduces the required effort on the users' side to select a reasonable value for the number of approximate integration time t.

Let $\mathscr{W}_s$ be the posterior samples of $\mathscr{W}$ after discarding warmup iterations, for $s \in \{1, ..., S\}$.

Since the NN is over-parameterized and individual weights $W_{ij}^{(l)} \in \mathscr{W}$ are usually unidentified, the posterior distribution of the weights themselves might not be very meaningful. However, the samples can produce estimates of meaningful quantities such as the conditional quantile function (QF) $Q(\tau|\boldsymbol{X}, \mathscr{W})$. For example, the conditional QF estimator is the posterior mean

$$\hat{Q}_{\text{MCMC}}(\tau|\boldsymbol{X}, \mathscr{W}) = \frac{1}{S}\sum_{s=1}^{S} Q(\tau|\boldsymbol{X}, \mathscr{W}_s)$$

and point-wise credible bands are obtained as the sample quantiles of $Q(\tau|\boldsymbol{X}, \mathscr{W}_s)$.

3.6 The SPQR package

The R package **SPQR** has two model fitting functions, `SPQR()` and `cv.SPQR()`, as well as various helper functions for handling tasks such as model validation, model prediction, and results visualization. Table 3.2 lists all the main functions provided by **SPQR**.

3.6.1 The main fitting function

The `SPQR` function specifies a semiparametric conditional density regression model of the type (3.2) and estimates the model parameters using one of the four computational approaches described in the previous section. It has the following arguments

```
SPQR(X, Y, n.knots=10, n.hidden=c(10), activation=c("tanh","relu"),
  method=c("MLE","MAP","MCMC"), prior=c("GP","ARD","GSM"),
  hyperpar=list(), control=list(), normalize=FALSE, verbose=TRUE,
  seed=NULL, ...)
```

This function takes two required arguments, a $n \times p$ covariate matrix (without intercept column) `X` and a response vector `Y`. The covariate matrix is expected to contain only numeric features, and that all categorical features are converted to numeric values in advance. The covariates are also recommended, although not required, to be normalized/standardized to have the same scale to stabilize gradient based optimization that will be used to estimate the parameters. The response vector, on the other hand, is required to take values between 0 and 1. We provide a `normalize` argument such that, when setting `normalize=TRUE`, all variables will be scaled to the unit interval using min-max transformation. Their original scales will be recorded to back-transform the estimated density and quantile function. By default, however, we set `normalize=FALSE` as we want the users to have full control on

Table 3.2: The overview of functions in package **SPQR**.

Function	Description
`SPQR()`	Main function of the package. Fits SPQR using the MLE, MAP, or MCMC method. Returns an object of S3 class `"SPQR"`, a list which includes the fitted model (`model`), the model configuration (`config`), the control parameters (`control`), the running time (`time`), the covariate matrix (`X`), the response vector (`Y`), as well as method-dependent training information.
`cv.SPQR()`	Fits SPQR using MLE or MAP method, and computes K-fold cross-validation (CV) error.
`createFolds.SPQR()`	Generate pre-computed CV folds.
`summary()`	Extracts and computes a list of summary information of a `"SPQR"` class object. Returns an object of S3 class `"summary.SPQR"`.
`print.summary()`	Prints the contents of a `"summary.SPQR"` class object in a user-friendly way.
`print()`	Computes and prints the summary information of a `"SPQR"` class object. Equivalent to `print.summary(summary())`.
`coef()`	Computes and returns the estimated spline coefficients $\theta_k(\boldsymbol{X}, \widehat{\mathscr{W}})$ of a `"SPQR"` class object.
`predict()`	Computes and returns the estimated PDF/CDF/QF of a `"SPQR"` class object.
`QALE()`	Computes and returns the quantile accumulative local effects (ALE) of a `"SPQR"` class object.
`plotEstimator()`	Computes and plots the estimated PDF/CDF/QF curves of a `"SPQR"` class object.
`plotGOF()`	Performs a visual goodness-of-fit test for the estimated conditional PDF using probability inverse transformation method.
`plotMCMCtrace()`	Show trace plot of the log-likelihood or a specified estimate of a `"SPQR"` class object fitted with `method="MCMC"`.
`plotQALE()`	Computes and plots the quantile ALE effects of a `"SPQR"` class object.
`plotQVI()`	Computes and plots the ALE-induced quantile variable importance measures of a `"SPQR"` class object.
`autoplot()`	A wrapper function that creates a user-specified plot for a `"SPQR"` class object by calling one of the plot functions above.

how variables are scaled, such as using prior information on the domains of the variables, etc.

The arguments `n.knots` and `n.hidden` are the number of basis functions, K, and the number of hidden neurons, V_l, that define the SPQR model in (3.2). **SPQR** uses the function `mSpline()` in the **splines2** package (Wang and Yan 2021) to construct the basis functions. We require setting `n.knots` to at least 5 as the model may severely underfit otherwise. The `n.hidden` argument accepts a vector of integers such that `n.hidden[l]` is the the number of hidden neurons in layer l for $l \in \{1, ..., L-1\}$. It should be noted that although `n.knots` and `n.hidden` are given default values for convenience, they should be tuned per application for optimal performance of the SPQR estimators. The argument `activation` corresponds to the hidden layer activation function ϕ in (3.1), and is set to `activation="tanh"` for hyperbolic tangent by default.

The argument `method` determines the computational approach to be used to estimate the model parameters $\mathscr{W}$. The MLE and MAP estimators are obtained by solving (3.5) and (3.9) respectively using gradient-based stochastic optimization. Specifically, we use the Adam optimizer in the **torch** package. The **torch** package is an R implementation of the open source machine learning platform – PyTorch (Paszke et al. 2019). It supports hardware acceleration for systems with a CUDA-compatible NVIDIA graphical processing unit (GPU), which **SPQR** takes advantage of. The HMC and NUTS algorithms are implemented using C++ and mirror those in Stan (Stan Development Team 2022). We did not use Stan directly for two reasons. First, Stan uses automatic differentiation whereas the analytical gradients of NNs are fairly easy to derive and evaluate. **SPQR** depends on **Rcpp** and **RcppArmadillo** to efficiently compute the log-posterior and its gradients. Secondly, Stan does not allow block-updating using both HMC and Gibbs sampler that exploits the conjugacy of hyper-priors. By default, we set `method="MAP"` since it is computationally faster than the MCMC method and less prone to overfit than the MLE method.

The argument `prior` is used only for the Bayesian methods and corresponds to one of the three variance hyperpriors described in Table 3.1. We set `prior="GP"` as default for simplicity. The argument `hyperpar` is a list of named hyper-prior hyperparameters to use instead of the default values, including `a_lambda`, `b_lambda`, `a_sigma` and `b_sigma`. They correspond to a_λ, b_λ, a_σ and b_σ in (3.8). The default value is 0.001 for all four hyperparameters.

The argument `control` is a list of named and method-dependent parameters that allows finer control of the behavior of the computational approaches. The available pa-

rameters for MLE and MAP estimators are shown in Table 3.3. The NNs for MLE and MAP estimators are structured using a templated module that consists of only fully-connected layers (`"nn_linear"`), batch normalization (`"nn_batch_norm1d"`), and dropout (`"nn_dropout"`). Dropout and batch normalization are not used by default, but may be useful when training deep and wide NNs. We recommend setting `use.GPU=TRUE` to further accelerate computation on CUDA-configured systems. Early stopping is implemented to avoid overfitting. We use `valid.pct`×100% of the data as the validation set. During each epoch, a snapshot of the trained model is saved in `save.path/save.name` if it leads to a decrease in the validation loss. When the validation loss stops decreasing for `early.stopping.epochs`, the training stops and the best model is loaded and returned.

Table 3.3: Control parameters for MLE and MAP.

Parameter	Description
`use.GPU`	A Boolean flag for GPU utilization. Default is FALSE.
`lr`	Learning rate of Adam optimizer. Default is 0.01.
`dropout`	Dropout probabilities. A length-two vector of which the first entry specifies the dropout probability in the input-to-hidden layer and the second entry specifies the probabilities in all hidden-to-hidden layers. Default is `c(0,0)` which corresponds to no dropout.
`batchnorm`	A Boolean flag for batch normalization. Default is `FALSE`.
`epochs`	The (maximum) number of passes of the entire training dataset by Adam. Default is 200.
`batch.size`	Size of mini batches to calculate gradient. Default is 128.
`valid.pct`	Percentage of data used as validation set. Default is 0.2.
`early.stopping.epochs`	The number of epochs before stopping if the validation loss does not decrease. Default is 10.
`print.every.epochs`	The number of epochs before next training progress in printed. Default is 10.
`save.path`	Path to save the trained torch model with the lowest validation loss. Default is `file.path(getwd(),"SPQR_model")`.
`save.name`	File name to save the trained `torch` model with the lowest validation loss. Default is `"SPQR.model.pt"`.

The available parameters for MCMC estimator are shown in Table 3.4. These parameters have similar meanings to those in the `stan()` function in the **rstan** package (Guo et al. 2021), and detailed explanations of their effects can be found in the Stan reference manual (Stan Development Team 2022). By default, **SPQR** uses `algorithm="NUTS"` to approximate the posterior distribution of $\mathscr{W}$ as it adaptively selects both the leap-frog discretization step-size ϵ as well as the number of steps L_ϵ which are crucial to the sampling efficiency of HMC. The value of step-size is also affected by the target Metropolis acceptance rate `delta`. When the NN is large and its posterior geometry is complex, a larger `delta` allows NUTS to explore the posterior more carefully using smaller steps. However, a smaller step-size may require a larger `max.treedepth` to avoid premature stopping which will significantly increase the algorithm run-time.

Finally, the argument `verbose` determines whether training progress should be printed, `seed` sets the seed for random number generation when reproducibility is desired, and `...` allows any of the control parameters to be specified directly in the function call instead of in `control`.

3.6.2 Cross-validation function

As seen above, the Adam routine used to compute MLE and MAP estimators has numerous potential tuning parameters, such as the learning rate and batch size. In addition, the number of basis functions and hidden neurons will also affect the quality of the estimator. To allow users to select the best model by comparing different model configurations, we provide the `cv.SPQR()` function that calculates the cross-validation (CV) error for a given configuration of the MLE or MAP estimator. In addition to all arguments of `SPQR`, `cv.SPQR()` takes the argument `folds` which are pre-computed folds on which CV can be performed. It should be noted that `cv.SPQR()` itself does not perform model selection, but can be used as a building block in a grid search loop. There are many ways to generate pre-computed CV folds, such as the `createFolds()` function in the **caret** package (Kuhn 2022). For users' convenience, we provide the function `createFolds.SPQR()` which is equivalent to `createFolds()` function in **caret** but returns an unnamed list. Among all control parameters listed in Table 3.3, we recommend tuning `lr`, in addition to `n.knots` and `n.hidden`. A reasonable range of values to consider is `n.knots` $\in \{8, 10, 12\}$, `n.hidden` $\in \{8, 10, 15\}$ and `lr` $\in \{e^{-6}, e^{-5}, e^{-4}, e^{-3}\}$.

The `cv.SPQR()` function is only applicable to MLE and MAP estimators. For the MCMC

Table 3.4: Control parameters for MCMC. These parameters are similar to those in `stan()` in **rstan**. Detailed explanations can be found in the Stan reference manual.

Parameter	Description
`algorithm`	The sampling algorithm; `"HMC"`: Hamiltonian Monte Carlo with dual-averaging, `"NUTS"`: No-U-Turn sampler (default).
`iter`	The number of iterations (including warmup). Default is 2000.
`warmup`	The number of warmup/burn-in iterations for step-size and mass matrix adaptation. Default is 500.
`thin`	The period for saving post-warmup samples. The default is 1.
`stepsize`	The discretization interval/step-size ϵ of leap-frog integrator. Default is `NULL` which means it will be adaptively selected during warmup iterations.
`metric`	The mass matrix **M**; `"unit"`: diagonal matrix of ones, `"diag"`: diagonal matrix with positive diagonal entries estimated during warmup iterations (default), `"dense"`: a dense, symmetric positive definite matrix during warmup iterations.
`delta`	The target Metropolis acceptance rate. Default is 0.9.
`max.treedepth`	The maximum tree depth in NUTS. The default is 6.
`int.time`	The integration time t in HMC. The number of leap-frog steps is then calculated as $L_\epsilon = \lfloor t/\epsilon \rfloor$. Default is 1.

estimator, prediction accuracy can be measured by the expected log pointwise predictive density (ELPD) which can be estimated by Bayesian leave-one-out (LOO) CV or by widely applicable information criterion (WAIC, Vehtari et al. 2017). Calculation of these statistics is integrated in the `summary()` functionality and described in the following section. Once convergence is achieved, we recommend running separate chains with different combinations of `n.knots` and `n.hidden` and select the best model with the highest ELPD.

3.6.3 Helper functions for "SPQR" object

The `SPQR()` function returns an object of S3 class `"SPQR"`, a compound list that contains the fitted model besides various other information. The `summary()` function summarizes the output produced by `SPQR()` and structures them in a more organized way to be examined by the user. For `SPQR()` fitted using any of the four methods, the output returned by `summary()` contains the estimation method, the run-time, and the NN architecture.

For MLE and MAP estimators, the output contains the training and validation loss of the final (best if early stopping is used) model, and selected information about the optimizer such as learning rate and batch size. For MAP and MCMC estimators, the output contains the variance hyper-prior model. For MCMC estimator, `summary()` calculates various diagnostic statistics that can be used to evaluate the fit of the model and convergence of the MCMC chain. Specifically, we calculate Bayesian LOO-CV and WAIC using the **loo** package (Vehtari et al. 2020) for model comparison, and the average Metropolis acceptance ratio and the number of divergences of post-warmup iterations to determine if the chain is reliable. The output of `summary()` is also an object of S3 class `"summary.SPQR"`, to which `print.summary()` can be applied to print the aforementioned contents in a user-friendly way. The function `print.summary()` has an optional argument `showModel` that when set to `TRUE`, additionally prints the NN architecture by layer. Instead of using `summary` and `print.summary`, the user can achieve the same goal by directly `print()` the `"SPQR"` object in one function call.

The `coef()` function outputs the estimated spline coefficients $\theta_k(\mathbf{X}, \widehat{\mathscr{W}})$ given `X`. For MCMC estimator, these will be the posterior means of $\theta_k(\mathbf{X}, \mathscr{W})$. The `predict()` function computes different estimates of the conditional distribution. When `type="PDF"` or `type="CDF"`, it computes $f(Y|\mathbf{X}, \widehat{\mathscr{W}})$ or $F(Y|\mathbf{X}, \widehat{\mathscr{W}})$ for every combination of `X` and `Y`; when `type="QF"`, it computes $Q(\tau|\mathbf{X}, \widehat{\mathscr{W}})$ for every combination of `X` and `tau`. The argument `Y` is optional, and when left unspecified `predict()` will use `nY` to define a grid on [0,1] for estimation. This is useful when the user wants to estimate the full PDF/CDF/QF curve. For MCMC estimator, two additional arguments are available: `ci.level` and `getAll`. The argument `ci.level` allows the user to extract `ci.level`×100% credible intervals of the estimates in addition to the posterior means, whereas setting `getAll=TRUE` allows the user to extract all posterior samples of the corresponding estimates.

3.6.4 Quantile accumulated local effect (QALE)

The function `QALE()` is largely based on the `ALEPlot()` function in the **ALEPlot** package (Apley 2018), but adapted for the QR setting to compute ALEs at different quantiles. The function takes the following arguments,

```
QALE(object, var.index, tau=seq(0.1,0.9,0.1), n.bins=40,
     ci.level=0, getAll=FALSE, pred.fun=NULL).
```

The argument `object` corresponds to the fitted SPQR object of class `"SPQR"`. The argument

`var.index` is a numeric scalar or length-two vector of indices of the covariates for which the ALEs will be calculated. When `length(var.index) = 1`, the function calculates the main effects for `X[,var.index]`; when `length(var.index) = 2`, the function calculates interaction effects for `X[,var.index[1]]` and `X[,var.index[2]]`. The argument `tau` is a numeric vector of quantile levels at which the ALEs will be calculated. The argument `n.bins` is a numeric scalar that specifies the maximum number of intervals into which the covariate range is divided when calculating the ALEs. The actual number of intervals depends on the number of unique values in `X[,var.index]`. When `length(var.index) = 2`, `n.bins` is applied to both covariates. The arguments `ci.level` and `getAll` allow uncertainty analysis of the calculated ALEs, but are only implemented for main effects for a single covariate. Finally, the argument `pred.fun` accepts a custom quantile prediction function that will be used instead of the built-in `predict()` function for calculating ALE. This can be useful when the user wants to compare the QALE calculated using SPQR to that using other QR models, or maybe that using the true model in a simulation study.

3.6.5 Plot functions

Various plot functions are implemented to visualize results from model prediction, model diagnostics and model interpretation. All of them take directly the `"SPQR"` object as input and return a `"ggplot"` object that can be further customized using layers from the **ggplot2** package (Wickham 2016). The `plotEstimator()` function computes and plots the estimated PDF/CDF/QF curve for a single observation. The `plotGOF()` function plots the quantiles of probability integral transform of the observed data against the quantiles of a uniform distribution. Let $U_i = F(Y_i|\mathbf{X}_i, \widehat{\mathscr{W}})$ be the estimated CDF of the ith observation in the dataset. By the probability integral transform, if the observed data indeed distribute according to $F(Y|\mathbf{X}, \widehat{\mathscr{W}})$ then the sample $U_i,\ i \in \{1, ..., n\}$ should correspond to independent samples of a standard uniform distribution, i.e., $U_1, ..., U_n \overset{iid}{\sim} \mathscr{U}(0,1)$. Thus the Quantile-Quantile plot (Q-Q plot) created by `plotGOF()` can be used for visually checking the goodness-of-fit of the SPQR estimator. When SPQR is fitted with `method = "MCMC"`, the function `plotMCMCtrace()` can be used to show trace plot of a `target` to examine the convergence and autocorrelation of the MCMC chain. Available `target` are `"loglik"` for the log-likelihood, and `"PDF"`, `"CDF"` and `"QF"` for corresponding estimate for a single observation. The function also takes an optional argument `window` which allows examination of different parts of the chain. The functions `plotQALE()` and `plotQVI()` support

the `QALE()` function in helping the user understand how the covariates affect different quantiles, and thus accept similar arguments. In particular, the function `plotQALE()` plots the estimated quantile ALE main effects when `length(var.index) = 1` and quantile ALE interaction effects when `length(var.index) = 2`, whereas the function `plotQVI()` compares the quantile ALE-induced variable importance (VI) between all considered covariates. The `var.index` argument can be left unspecified in `plotQVI()` in which case all covariates in the data set are considered. Similar to the `predict()` function, most plot functions accept arguments `ci.level` or `getAll` (or both) for uncertainty quantification when SPQR is fitted with `method = "MCMC"`. Table 3.5 summarizes the implementation of uncertainty quantification for plot functions in **SPQR**. Finally, the function `autoplot()` provides a wrapper for all the plot functions mentioned above.

Table 3.5: Implementation of uncertainty quantification for plot functions in **SPQR**.

Name of the function	`ci.level`	`getAll`
`plotEstimator()`	✓	✓
`plotGOF()`	×	✓
`plotMCMCtrace()`	N/A	N/A
`plotQALE()`	✓	✓
`plotQVI()`	✓	×

3.7 Examples

The R codes that produce the results shown in this section are available upon request.

3.7.1 Simulation

We start with a simple simulation study to demonstrate the effectiveness of SPQR in estimating conditional density and quantile functions, as well as usages of functions introduced in the previous section. We consider a three dimensional covariate $\boldsymbol{X} = (X_1, X_2, X_3)$ with variables independently generated from a uniform distribution. The response Y follows a Beta distribution whose shape parameters are functions of X_1 and X_2, the third covariate is

irrelevant.

$$\begin{aligned} X_j &\overset{iid}{\sim} \mathscr{U}(0,1),\ j=1,...,3 \\ Y &\sim \mathscr{B}\text{eta}(10\text{expit}\{1-5X_1X_2\}, 10[1-\text{expit}\{1-5X_1X_2\}]). \end{aligned} \tag{3.11}$$

Here $\text{expit}(u) = 1/(1+e^{-u})$ is the inverse logistic link function. Based on the definition of Beta distribution, the following properties of the conditional distribution $f(Y|\boldsymbol{X})$ can be obtained

$$\begin{aligned} \mathbb{E}(Y|\boldsymbol{X}) &\propto \text{expit}\{1-5X_1X_2\} \\ \text{Var}(Y|\boldsymbol{X}) &\propto \text{expit}\{1-5X_1X_2\}[1-\text{expit}\{1-5X_1X_2\}] \\ \text{Skewness}(Y|\boldsymbol{X}) &\propto -\frac{\sqrt{\text{expit}\{1-5X_1X_2\}}}{\sqrt{1-\text{expit}\{1-5X_1X_2\}}}. \end{aligned}$$

That is, the location, scale and shape of $f(Y|\boldsymbol{X})$ all depend on $\boldsymbol{X}$. Fig. 3.1 shows the true density and quantile functios of Y as conditioned on different combinations of X_1 and X_2. We can clearly observe a varying effect of $\boldsymbol{X}$ on the density and quantile functions of Y.

We generate $n = 1000$ samples from (3.11) and fit SPQR using the MLE, MAP, and MCMC methods. For all three methods, the default configuration of `n.knots=10`, `n.hidden=10` and `activation="tanh"` are used. For the MLE and MAP methods, the learning rate is selected from $\{e^{-6}, e^{-5}, e^{-4}, e^{-3}\}$ using 5-fold cross-validation, and the gradients are calculated using mini-batches of size 256 with a maximum training time of 500 epochs. For the MAP method, the default `prior="ARD"` is used. For the MCMC method, the posterior distribution is approximated using NUTS. We run NUTS for a total of 1000 iterations, discard the first 250 as warm-ups, and save every other iteration.

The function `print()` returns a short summary of the results of the fitted object. The summary depends on the fitted method. Here we show summary for models fitted with `method="MLE"` and `method="MCMC"`. The summary for `method="MAP"` is mostly the same as that for `method="MLE"` and thus omitted.

```
#> SPQR fitted using MLE approach
#>
#> Learning rate: 0.04978707
#> Batch size: 256
#>
#> Loss:
#>   train = -140.2645,  validation = -142.1546
```

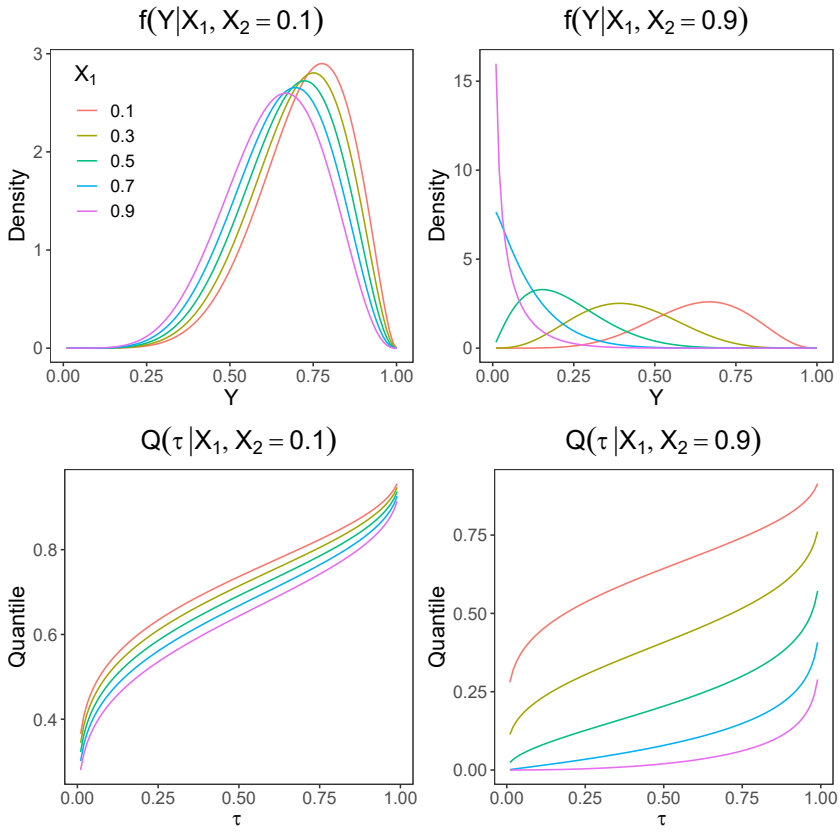


Figure 3.1: True conditional density and quantile functions of simulated Beta outcome for different combinations of X_1 and X_2.

```
#>
#> Elapsed time: 0.22 minutes

#> SPQR fitted using MCMC approach with ARD prior
#>
#> MCMC diagnostics:
#>   Final acceptance ratio is 0.91 and target is 0.9
#>
#> Expected log pointwise predictive density (elpd) estimates:
#>   elpd.LOO = 683.4493,  elpd.WAIC = 683.9026
#>
#> Elapsed time: 3.05 minutes
```

An overview of the NN architecture can be additionally printed by setting the argument `showModel` to `TRUE`.

```
#> SPQR fitted using MLE approach
#>
#> Learning rate: 0.04978707
#> Batch size: 256
#>
#> Model specification:
#>     Layers
#>    Input Output Activation
#>        3     10       tanh
#>       10     10    softmax
#>
#> Loss:
#>   train = -140.2645,  validation = -142.1546
#>
#> Elapsed time: 0.22 minutes
```

The function `plotGOF()` uses a Q-Q plot to inspect the alignment of the probability integral transform (PIT) and the uniform distribution, which can be used to visually check the goodness of fit of `SPQR()`. Fig. 3.2 shows the goodness of fit of the three SPQR estimators. For SPQR fitted with `method="MCMC"`, the argument `getAll` is used to additionally plot all posterior samples of the Q-Q plot. The plots show that the distribution of PIT is most similar to the uniform distribution for the MCMC estimator, suggesting that the model based on the MCMC method best fits the observed data. The reliability of the MCMC estimator can be further assessed by the function `plotMCMCtrace()`, which plots traceplots that can be used to visually inspect the sampling behavior and convergence of the post-warmup chain. Fig. 3.3 provides results based on two examples of such usage: one uses `target="loglik"` to plot the traceplot of the log-likehood; the other uses `target="QF"`, `X=c(0.5,0.5,0.5)` and `tau=0.5` to plot the traceplot of the estimated median when $\boldsymbol{X} = (0.5, 0.5, 0.5)^\top$. In both cases visual evidence indicate that the chain converged.

To investigate the numerical performance of the fitted SPQR models, we use the function `predict()` to compute the predicted conditional PDFs and QFs for 3 out-of-sample observations and compare them with their true values. The estimated and true PDFs are shown in the top row of Fig. 3.4, and the estimated and true QFs are shown in the bottom row. For the MCMC estimator, we also compute the 95% pointwise credible intervals using the argument `ci.level=0.95` and plot them in Fig. 3.5. These plots show that SPQR successfully captured the varying effect of $\boldsymbol{X}$ on the distribution of Y. Fig. 3.4 and Fig. 3.5 can be easily reproduced using `plotEstimator()` which inherits most arguments of `predict()`.

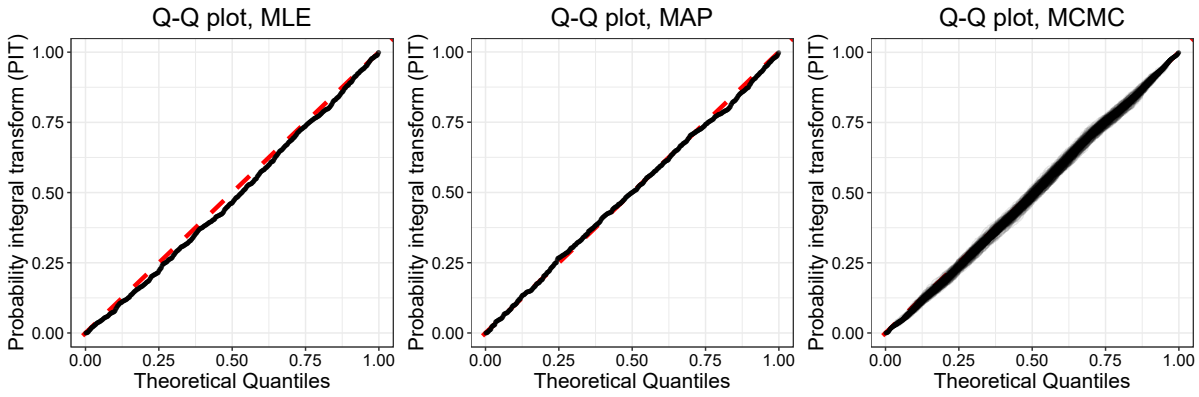


Figure 3.2: Examples of plot produced by `plotGOF()`. Goodness-of-fit test for SPQR estimators.

In most applications where QR is used, understanding the covariate effect on the conditonal quantile/distribution is of paramount importance. The function `QALE()` can be used to quantify either the main effect of a single covariate or the interaction effect between two covariates on the predicted quantiles. For illustration, we compute the QALE for X_1, X_2, X_3 respectively at $\tau = 0.25$ using fitted `"SPQR"` objects, as well as a custom prediction function (in this case it is the true quantile function). To compute the QALE using a custom prediction function, the user should first define a function that takes `X`, the covariate matrix, and `tau`, a vector of quantile levels, as inputs and returns an `nrow(X)` by `length(tau)` matrix of predicted quantiles. The user should then pass the defined function to the `pred.fun` argument and `list(X=X)` to the `object` argument. The results are shown in Fig. 3.6 and provide visual evidence that SPQR accurately captures the quantile covariate effects. We also compute the QALE interaction effect between X_1 and X_2 at $\tau = 0.5$. The results are shown in Fig. 3.7. The plots show that SPQR is able to recover the complex interaction effect between X_1 and X_2.

The function `plotQVI()` computes the QALE-induced variable importance (VI) measures of each covariate. The covariates are then ranked accordingly in a barplot at each quantile of interest. Fig. 3.8 shows the quantile VI for X_1, X_2 amd X_3 at $\tau \in \{0.1, 0.5, 0.9\}$, respectively. The argument `ci.level = 0.95` is used to plot 95% error bar for each VI measure. The result suggests that X_1 and X_2 have similar and significant effect on quantiles of Y whereas X_3 has no effect. The effects of X_1 and X_2 are also more prominent on upper quantiles than on lower quantiles. These observations match the truth.

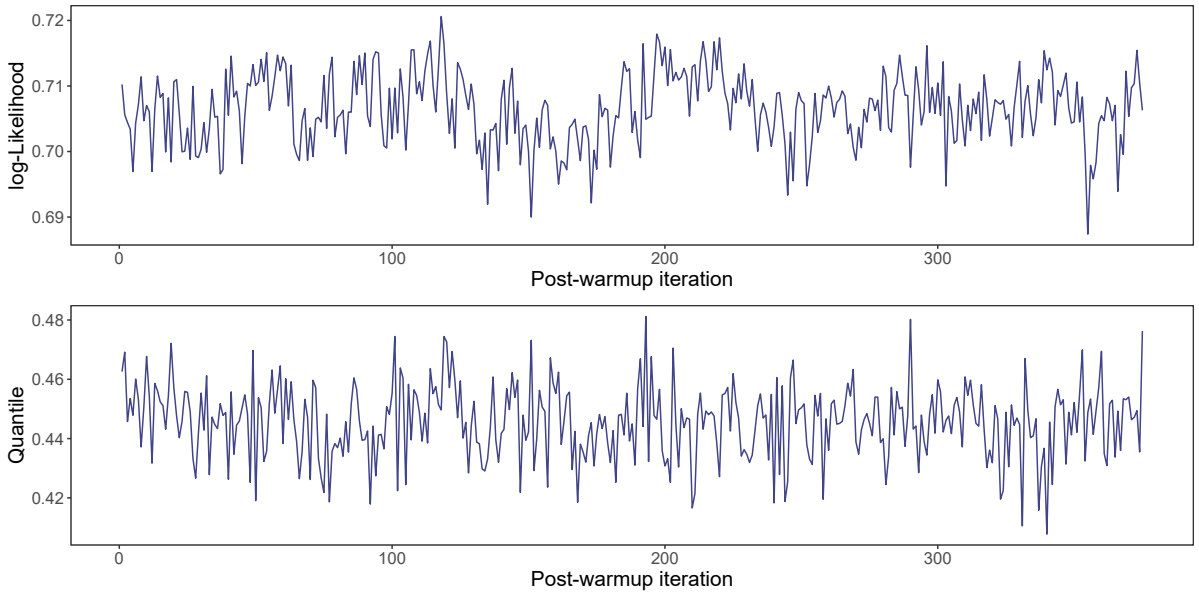


Figure 3.3: Examples of traceplot produced by `plotMCMCtrace()`.

3.7.2 Australia electricity demand data

For real data application we consider the electricity demand data from Sydney, Australia. The dataset contains electric energy consumption, recorded from smart meters, of 247 anonymized residential customers during the period between July 3rd 2010 and June 30th 2011. It is available from the R package **qgam** and has been analyzed by Fasiolo et al. (2021) in the context of QR. In particular, we are interested in analyzing the effect of temperature and time on the average demand distribution and its different quantiles. Hence the response variable is chosen to be `dem`, and the set of covariates contains `doy`, `tod`, `temp` and dummy variables representing `dow`.

Given that the sample size is fairly large, we consider fitting SPQR using the MLE method. The modeling parameters are selected using a grid search. Specifically, we select the number of basis functions (`n.knots`) from $\{10, 15, 20\}$ and the number of hidden neurons (`n.hidden`) from $\{10, 15, 20\}$. We focus on 2-hidden-layer neural networks so `n.hidden` represents the number of neurons in each hidden layer. For the control hyperparameters, we also select learning rate (`lr`) from $\{e^{-3}, e^{-4}, e^{-5}\}$; we set the batch size (`batchsize`) to be 128, maximum number of epochs (`epochs`) to be 800, and early stopping criterion (`early.stopping.epochs`) to be 50. The best model configuration is selected based on

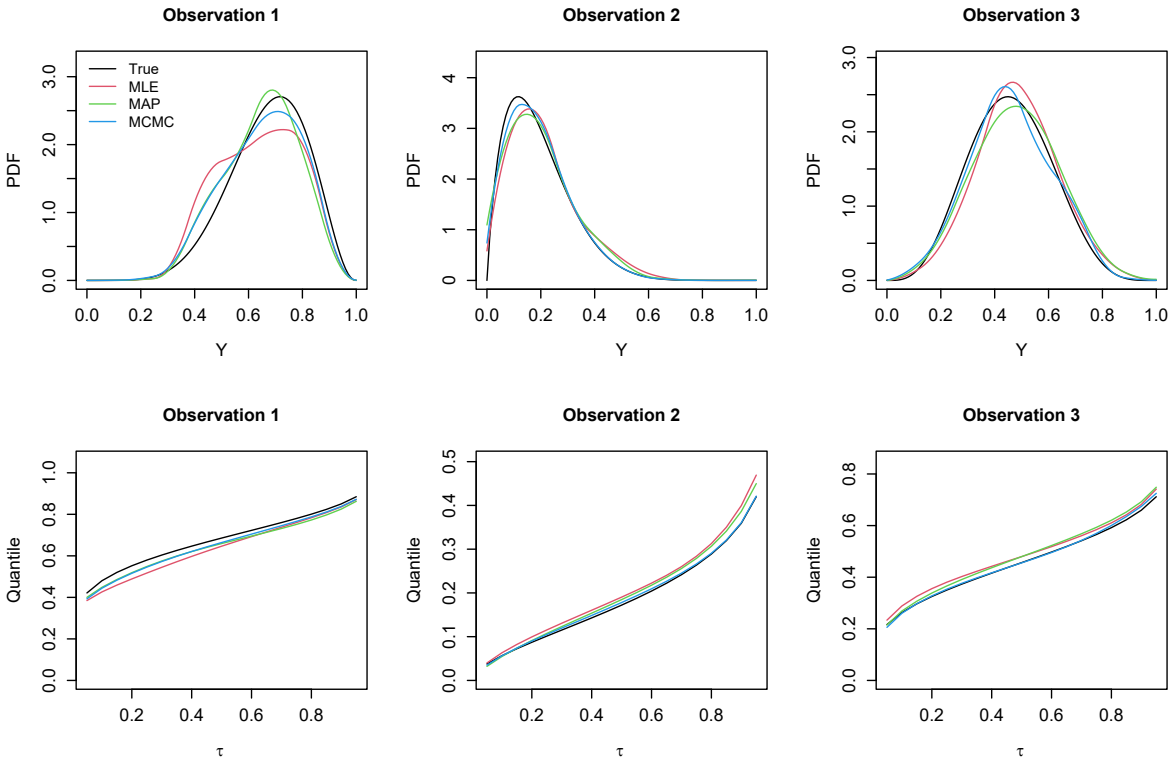


Figure 3.4: Estimated and true PDFs (top) and QFs (bottom) for 3 out-of-sample observations.

10-fold CV error.

To accelerate the grid search routine, the function `foreach()` from the **foreach** package (Microsoft and Weston 2022) can be used to run jobs in parallel. However, since tensor computation in the **torch** package is very memory-consuming, using many cores can easily exceeds the memory limit. In general, we recommend the user to start with 2 cores and then adjust accordingly to the available memory. After CV, We refit the model with the best configuration, using 10% of data as validation set for early stopping criterion.

To determine which covariates are important in predicting quantiles of `dem`, we compare the quantile VI of all covariates at $\tau \in \{0.1, 0.5, 0.9\}$. The result is shown in Fig. 3.9. Across all three quantiles, `doy`, `temp` and `tod` have the highest VI, suggesting strong seasonal and hourly effects. These effects are also more prominent on upper tail of the average demand distribution than on central region and lower tail. While one might expect that there is also a daily/weekend effect on electric consumption, the estimated effect of `dow` seems to be almost negligible, suggesting no particular difference in terms of electric consumption

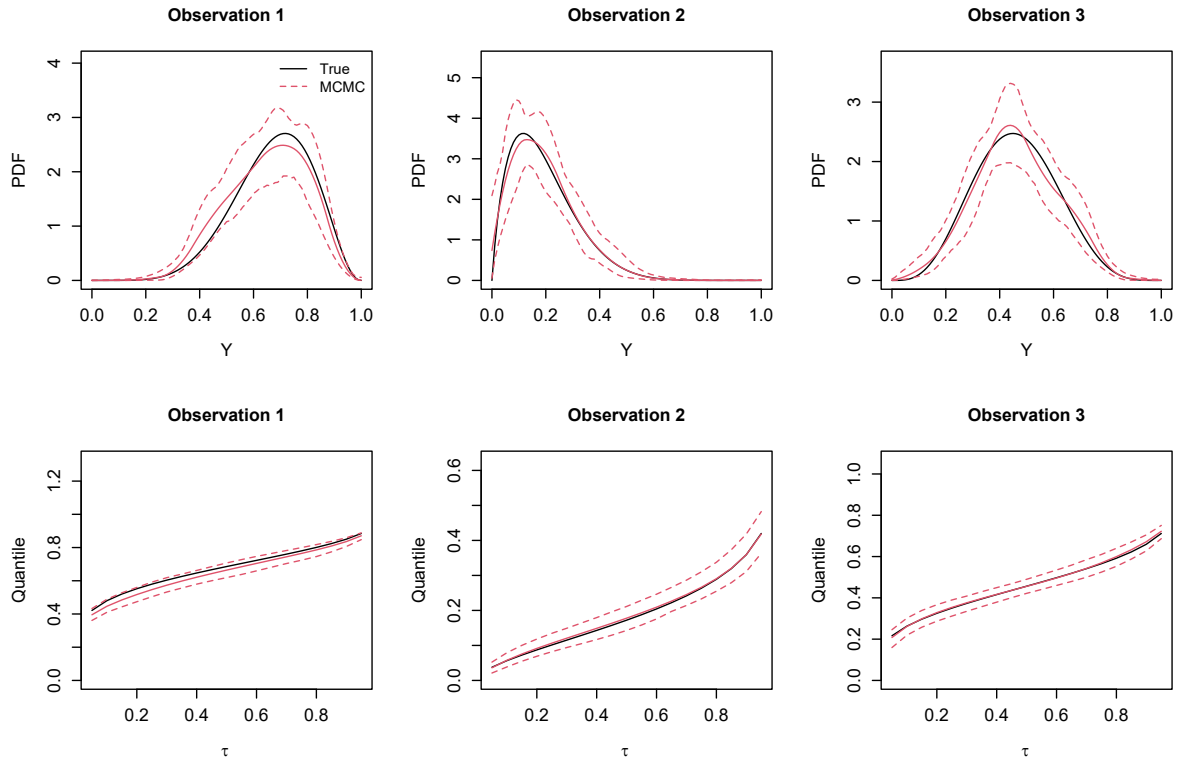


Figure 3.5: Estimated and true PDFs (top) and QFs (bottom), along with 95% credible bands, for 3 out-of-sample observations.

across the week. One reason could be that only data between 17:30 and 21:30 of the day are used, in which case the probability of people being home is similar between weekdays and weekends.

To further investigate the effects of `doy`, `temp` and `tod` on the average demand distribution, we compute their quantile ALE main effects at $\tau \in \{0.1, 0.5, 0.9\}$. The results are plotted in Fig. 3.10. The estimated effect of `doy` has a unimodal shape at $\tau = 0.1$ and $\tau = 0.5$ and peaks at austral winter. Suggesting a high electric consumption when the weather is at its coldest. For $\tau = 0.9$, the effect also has a minor mode at austral summer, suggesting that the average demand distribution is highly right-skewed during austral summer. The estimated effect of `tod` displays a quadratic upwarding trend before 20:00 and a linear downwarding trend afterwards. This suggests that use of electricity becomes more active as people start to get home and becomes less active as the night closes in. The downwarding trend is also more prominent at upper tail of the average demand distribution than at central region or

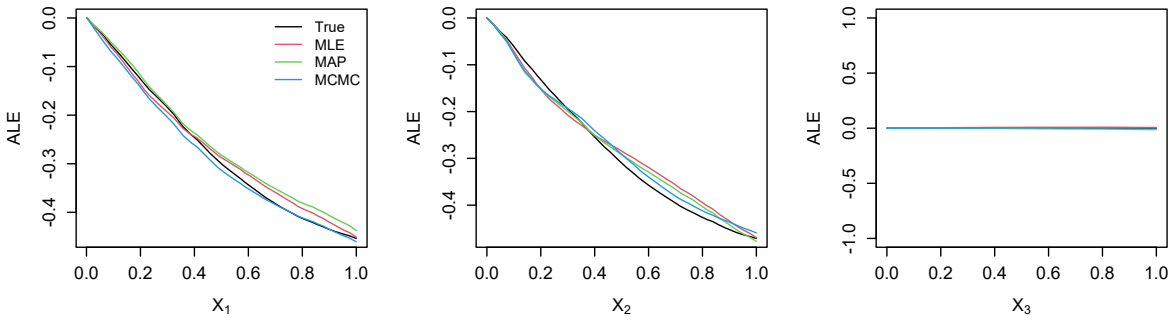


Figure 3.6: Quantile accumulative local effects (ALEs) for X_1, X_2 and X_3 respectively at $\tau = 0.5$.

lower tail. This suggests that as the day ends and people start to go to bed, the probability of very high electric consumption significantly decreases. The estimated effect of `temp` seems to complement that of `doy`. It has a "check" shape and shows a significant upwarding trend when temperature is above 20 degree. This suggests a significantly higher electric consumption when cooling becomes necessary during summer. We also notice that the turning point for `temp` effect at upper quantiles is closer to 15 degree than at lower quantiles. This may suggest that people who contribute to very high electric consumption tend to turn on cooling at a lower temperature than those who contribute to very low electric consumption. Similar characteristics of these effects were also observed in Fasiolo et al. (2021).

3.8 Discussion

In this chapter, we present the R package **SPQR** for fitting the semiparametric conditional density and QR models as proposed in Xu and Reich (2021). The main advantage of these models is their capability of modeling complex covariate effects on the response distribution, which is absent in existing parametric QR models. Furthermore, the estimated distribution and quantile functions are always valid, ensuring sensible inference even when the sample size is small. The package also provides a framework for fully Bayesian inference of the fitted models to allow uncertainty quantification, as well as model agnostic tools to understand the effects of different covariates on different parts of the distribution when model transparency is needed. We hope that this package will be a valuable addition to the existing family of QR tools and be especially useful for analyzing complex and possibly

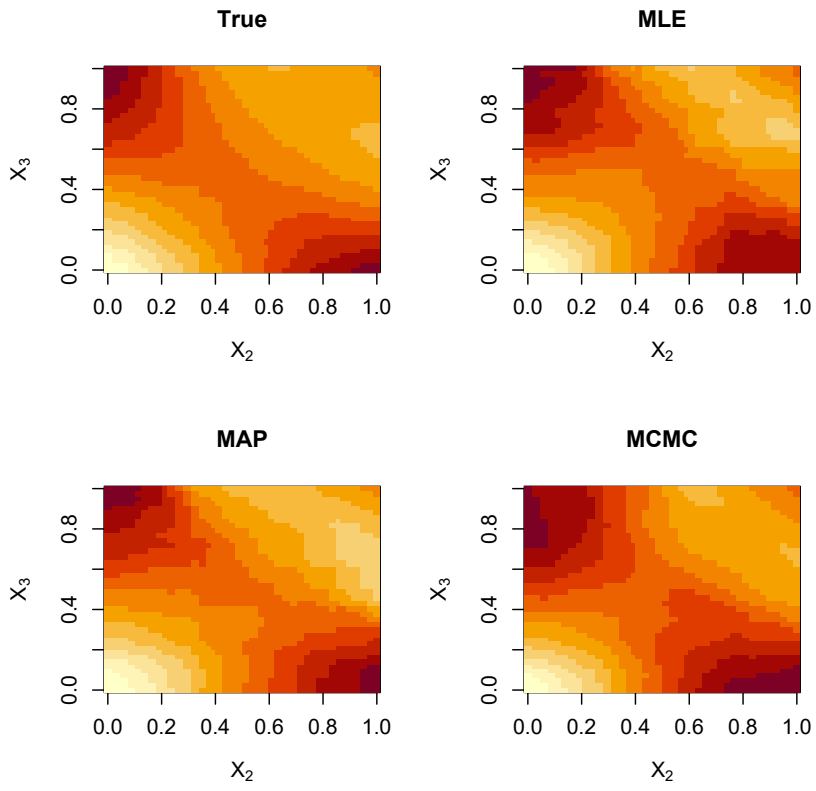


Figure 3.7: Quantile accumulative local effects (ALEs) interaction effect between X_1 and X_2 at $\tau = 0.5$.

high-dimensional heteroscedastic data where flexible density and QR models are suitable. Future work will aim at improving the scalability of the Bayesian SPQR fitting framework by enabling sparsity-induced priors and implementing variational Bayes estimator. We also plan to include functions for more specialized use case, such as estimation of counterfactual distribution and quantile treatment effects in a causal inference context.

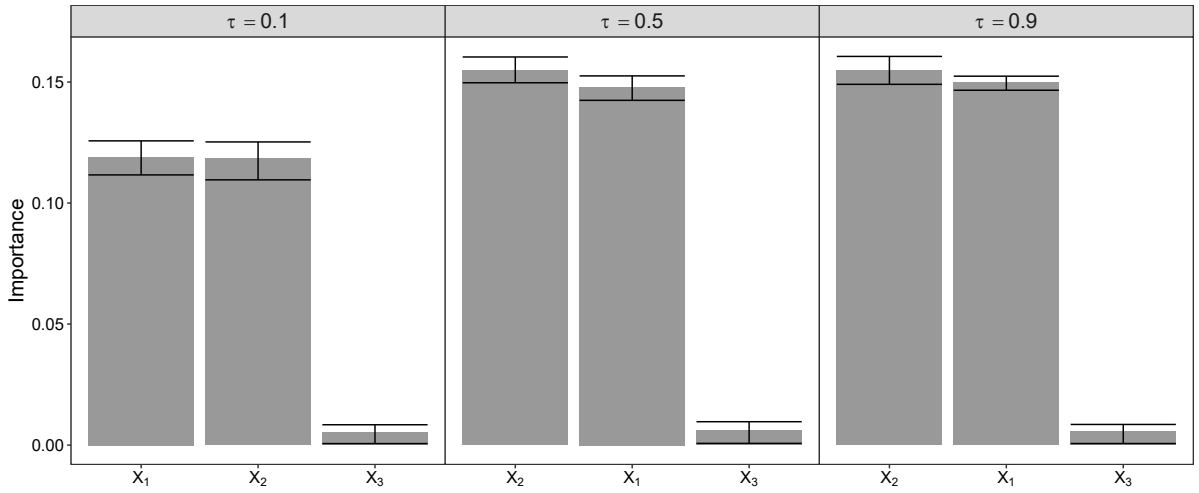


Figure 3.8: Example of plot produced by `plotQVI()`. Quantile variable importance (VI) at $\tau \in \{0.1, 0.5, 0.9\}$.

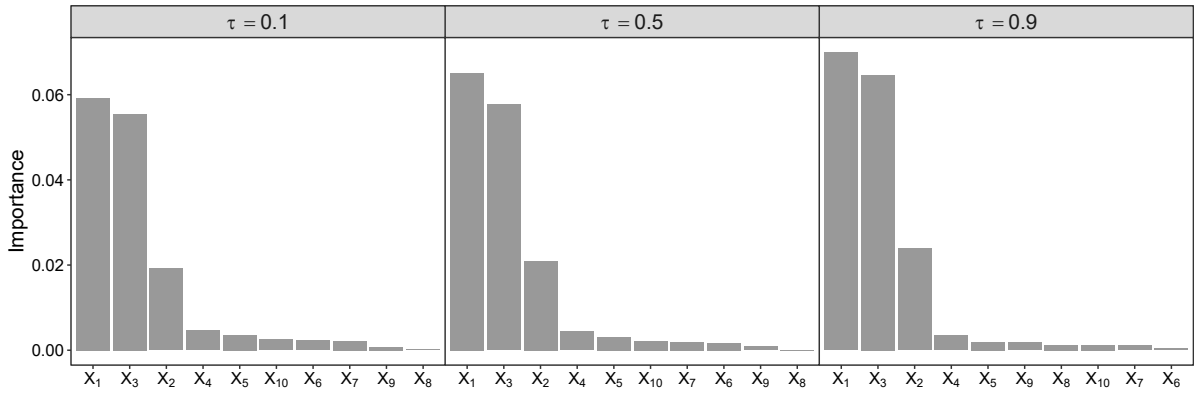


Figure 3.9: Variable importance (VI) of Australia electricity demand data. Estimated quantile VI of all covariates at $\tau \in \{0.1, 0.5, 0.9\}$.

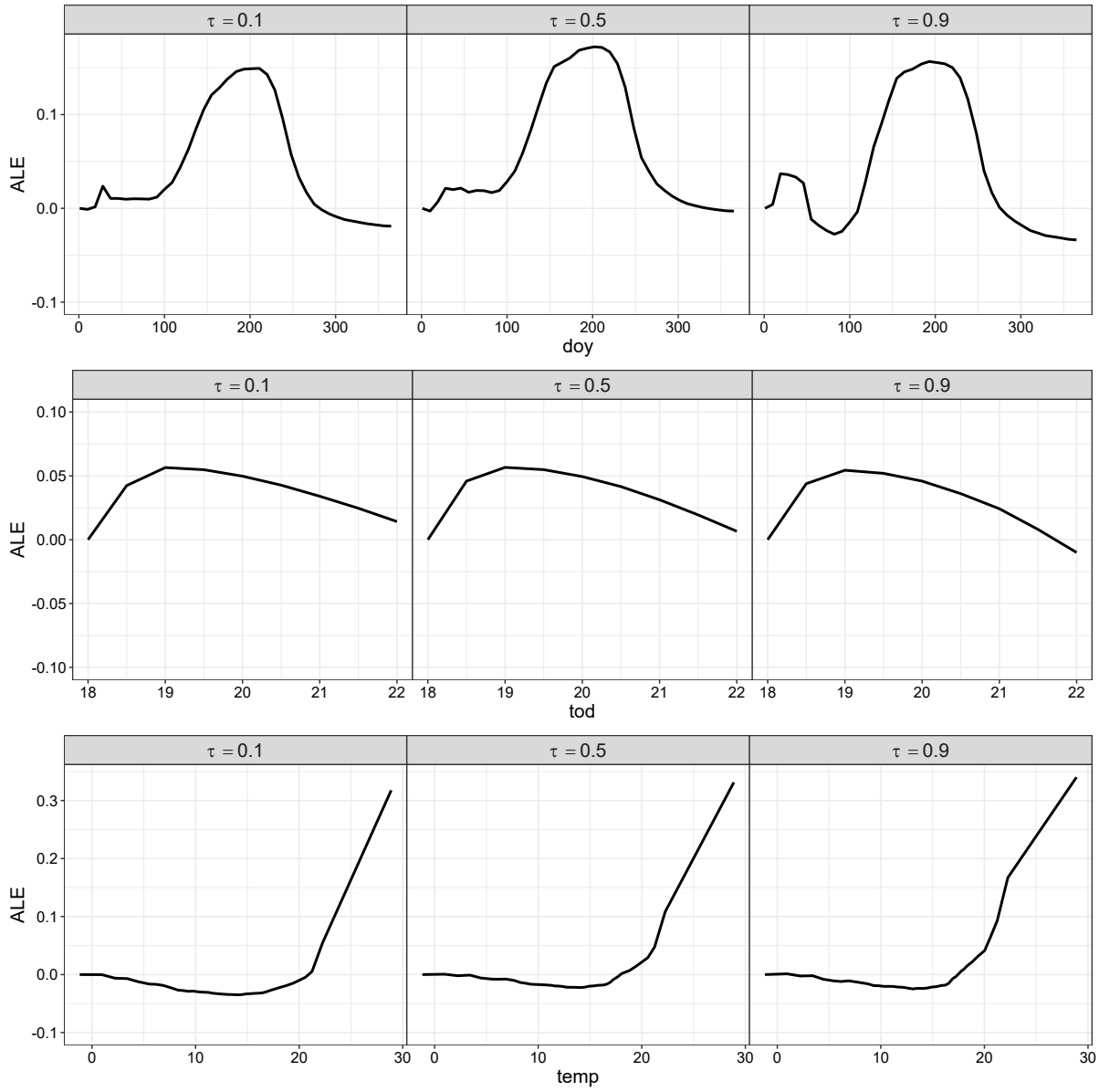


Figure 3.10: Estimated quantile ALE main effects of day of the year (`doy`), time of the day (`tod`) and temperature (`temp`) at $\tau \in \{0.1, 0.5, 0.9\}$.

CHAPTER

4

A BAYESIAN SEMIPARAMETRIC METHOD FOR ESTIMATING CAUSAL QUANTILE EFFECTS

The work in this Chapter has been submitted as a paper under the same name.

4.1 Background

Statistical methods for estimating causal effects of a treatment or exposure using observational data play a central role in many research fields, e.g., social and economic science (Imbens and Rubin 2015), health care (Xie et al. 2020), public policy (Frölich and Melly 2013) and environmental science (Sun et al. 2021). Most of the existing methods are cast in terms of the potential outcomes framework (Rubin 1978). Under this framework, causal inference can be drawn by comparing suitable functions of the distribution of *counterfactual* or *potential* outcomes, defined as the outcomes that would have been observed for

the subjects if they were assigned treatment or control.

The potential outcomes framework emphasizes the importance of randomization. In a randomized trial, the distribution of potential outcomes and their causal effects can be estimated using samples of the observed outcomes in the treatment and control groups. In an observational study, however, the treatment and control groups may be systematically different by some extraneous determinants of the outcome, or confounders, that contribute to the difference in both the treatment and outcome. Consequently, the sample difference becomes a biased estimate of the treatment effect, and a simple comparison of outcomes between the treatment and control group will not lead to valid causal interpretation.

To obtain consistent estimators of causal effects in observational studies, strong ignorability of treatment assignment (i.e., conditionally independent of the potential outcomes given the confounders) is commonly assumed. Under the strong ignorability assumption, causal effects can be estimated using the observed outcome after appropriate adjustment for confounders.

Much of the causal inference literature to date focuses on examining the impact of a treatment on the central tendency of the counterfactual distributions, and most papers quantify such impact using the average treatment effect (ATE) or the conditional average treatment effect (CATE). In practice, however, asymmetric distributions are frequently encountered and are better summarized in quantiles. Furthermore, the counterfactual distributions under treatment and control could be different in not only central tendency but also spread or shape. In such cases, the ATE itself might not be enough or even fails to characterize the distributional differences. In contrast, comparing different parts of or the entire counterfactual distributions can provide more nuanced and valuable measures for exploring causal effects beyond a mean shift.

Quantile treatment effects (QTE) (Doksum 1974) measure the differences between quantiles of potential outcome. It can capture heterogeneous causal effects of the treatment at different locations of the counterfactual distribution. There exist two types of QTE estimand. The *unconditional* QTE is defined as the difference between the quantiles of the counterfactual distributions of the treatment and control responses, marginalized over the distribution of confounders. This estimand is desirable when the ultimate objective is to examine the difference between distributions of the potential outcome that would be observed if the entire population were to receive the treatment versus the entire population were to receive the control, and can be intuitively interpreted as the “horizontal distance" between counterfactual distributions. The *conditional* QTE (CQTE), in contrast, measures

the difference between quantiles of potential outcome conditional on specific realization of observed covariates. It quantifies the impact of a treatment on certain quantiles of the counterfactual distribution for a sub-population defined by the conditioning covariates, and provide insight on the interaction between treatment and individual characteristics. In most causal inference applications, estimation of QTE rather than CQTE is of primary interest since effectiveness of a policy or a treatment is often justified by its global effect irregardless of individual heterogeneity. However, as studies in personalized policy and precision medicine gain increasing popularity, the CQTE could be a power tool alongside CATE for exploring the heterogeneity of treatment effects.

Counterfactual quantiles can be estimated by either minimizing a weighted check loss or inverting an estimate of the cumulative distribution function (CDF) of the potential outcomes. In the latter case, the CDF can be naturally estimated using empirical expectation of thresholded outcomes, and therefore estimation of counterfactual quantiles largely reduces to estimation of counterfactual mean. A large number of existing works in QTE estimation consider estimators of these two types (e.g., Firpo 2007; Rothe 2010; Zhang et al. 2012; Donald and Hsu 2014; Yang and Zhang 2020; Sun et al. 2021). A drawback of these methods is that different quantiles have to be estimated separately, which could result in a non-monotonic estimate of the quantile function when the sample size is small. In addition, the discrete nature of these estimators forbids inference on the counterfactual probability density function (PDF). The PDF can reveal potentially interesting characteristics of the counterfactual distribution that cannot be revealed by the CDF, such as multimodality, and are often more visually interpretable to practitioners.

Density estimation is a statistically more complex problem than CDF estimation even in the non-counterfactual case. Consequently, counterfactual density estimation has received far less attention than its counterpart. Some early attempts are DiNardo et al. (1996) and Robins and Rotnitzky (2001) which discussed possible estimators but did not proceed in full analysis. Recently, Kim et al. (2018) adopted the kernel estimator from Robins and Rotnitzky (2001) and proposed a doubly robust like estimator that uses inverse probability weighting (IPW) for bias correction, but the resulting estimate is highly non-smooth. Kennedy et al. (2021) proposed a similar estimation procedure, but uses a plug-in estimator based on truncated cosine series. However, extreme estimates of the propensity score (PS) as well as oscillating nature of the cosine series may result in an estimator with large variance.

There has been a growing interest in adapting Bayesian semi/nonparametric models to flexibly model the counterfactual distribution and estimate associated causal effects. Xu

et al. (2018) proposed a two-stage approach for estimating the counterfactual distribution. First, the PS is modeled using probit regression and Bayesian additive regression trees (BART; Chipman et al. 2010). For each treatment group, the counterfactual distribution conditional on the estimated PS is then modeled using a Dirichlet process mixture (DPM) of normals. Conditioning on the PS is an attractive solution to circumvent the curse of dimensionality while adjusting for confounding. However, the estimator might not be efficient enough when the outcome-PS model does not sufficiently characterize the outcome-covariate relationship.

4.2 Contribution

In this chapter, we propose a novel Bayesian semiparametric model for estimating the counterfactual distributions that allows inference on any functionals of the counterfactual distributions, including but not limited to counterfactual densities and quantile causal effects. Specifically, we model the counterfactual distributions by first modeling the conditional distributions of the outcomes given treatment and balancing score and then marginalizing it over the population distribution of the balancing score. Adjustment for balancing score is crucial in observational studies to reduce bias due to confounders. To formulate an efficient estimator that provides reliable inference of the counterfactual distributions, we propose to adjust for a double balancing score that augments the PS with individual covariates. To avoid the complexity due to posterior inference on the joint likelihood of the PS and the outcome (Zigler et al. 2013), we adopt a sequential approach that separately estimates PS and the outcome in two stages. First, the PS is estimated using BART probit. Then, the balancing score containing posterior sample of the PS is used to estimate the outcome distribution. To ensure a flexible counterfactual density estimator that adapts to skewness, heavy-tailedness and multimodality, we extend the semiparametric quantile regression (SPQR) model by Xu and Reich (2021) to model the conditional outcome distribution using a finite mixture of shape-constrained splines, where the mixing weights are modeled by neural networks (NN). Observational studies often involve high-dimensional covariates. To improve the scalability of the proposed method and regularize its complexity, we give the NN weights Gaussian scale mixture (GSM) priors to automatically determine relevant features and encourage network sparsity.

The proposed method differs from existing works in the following aspects.

1. Most existing works adjust for either the full-vector of covariates or the scalar PS (Imai et al. 2008; Vansteelandt and Daniel 2014; Xu et al. 2018). Adjusting for covariates generally leads to smaller variability of outcome residual compared to adjusting for the PS. Adjusting for the PS alleviates the curse of dimensionality due to high-dimensional regression. The proposed double balancing score approach takes advantage of both approaches. It makes full use of the observed information by aggregating signals from both the covariates as well as the treatment assignment mechanism. The intuition is similar to that behind doubly robust methods. If the PS as a function of covariates does not summarize the outcome mean model well, the inclusion of individual covariates can help control the variability of outcome residual after adjusting for the PS. On the other hand, if the outcome-covariate relationship is complex and not well approximated, inclusion of the PS can reduce residual confounding after adjusting for individual covariates.

2. Double score adjustment is different from doubly robust weighting methods that augments an IPW estimator with information from the outcome model (Zhang et al. 2012; Kennedy et al. 2021) or doubly robust matching based on matching a double score that includes a PS and a prognostic score (Yang and Zhang 2020). By incorporating the PS as a regressor, we alleviate the associated drawbacks of IPW such as high variance due to subjects with extreme PS and associated inefficiency of matching.

3. The estimation problem is substantially different to that in Xu and Reich (2021) where the focus is solely on conditional distribution and its quantiles. In the current context, the conditional distribution is an intermediate estimand used for estimating the marginal distribution and its quantiles. In addition, as we mention in Section 4.5.3, we account for the uncertainty of the distribution of covariates through Bayesian bootstrap (Rubin 1981).

4. The proposed approach provides estimates of the full conditional counterfactual distributions given covariates which are unavailable from PS-based estimators (Xu et al. 2018). These estimates have benefits for personalized medicine and allows one to assess the heterogeneity of treatment effect on individuals (Lu et al. 2018).

4.3 Organization

The rest of the chapter is organized as follows. Section 4.4 defines the causal estimands and states key identifying assumptions. Section 4.5 describes the double balancing score approach and the Bayesian semiparametric model to estimate counterfactual densities and quantile causal effects. In Section 4.6, we compare the performance of the proposed approach with other counterfactual distribution estimators for a wide range of simulated data. We analyze the QTE of smoking on low birth weight using the North Carolina birth weight data in Section 4.7 and conclude in Section 4.8.

4.4 Preliminaries

Let $T \in \mathscr{T}$ denote the treatment variable. For simplicity, we assume the treatment is binary, i.e., $\mathscr{T} = \{0,1\}$, such that $T = 0$ is the control treatment and $T = 1$ is the active treatment; the methods proposed in later sections extend naturally to multivalued treatments. Under the potential outcome framework, each level of treatment t corresponds to a potential outcome $Y(t)$, representing the outcome that would have been observed for an individual if they were assigned treatment $T = t$. In real applications, $\{Y(0), Y(1)\}$ cannot be simultaneously observed, and only one corresponds to the actual outcome Y. Let $\boldsymbol{X} = (X_1, ..., X_d)^\top \in \mathbb{R}^d$ denote a d-vector of exogenous covariates. The observed data consist of $(\boldsymbol{X}_i, T_i, Y_i)$, $i = 1, ..., n$, which we assume to be random samples of the joint distribution of $(\boldsymbol{X}, T, Y)$. In addition, let $\pi(\boldsymbol{X})$ be the propensity score (PS), defined as the probability of receiving active treatment given confounders, i.e., $\pi(\boldsymbol{X}) = P(T = 1|\boldsymbol{X})$.

To estimate the quantile causal effect, we make the following identifying assumptions.

Assumption 1 (Stable unit treatment value assumption, SUTVA)**.** *The potential outcomes for one subject is independent of the treatment assignment of others.*

Assumption 2 (Consistency)**.** *For any subject, the observed outcome given the assigned treatment is equal to the potential outcome:* $Y_i = T_i Y_i(1) + (1 - T_i) Y_i(0)$.

Assumption 3 (Strongly ignorable treatment assignment, SITA)**.** *The treatment assignment is independent of the potential outcomes given the confounders:* $\{Y(0), Y(1)\} \perp T|\boldsymbol{X}$.

Assumption 4 (Overlap)**.** *For any given values of confounders, the probability of assignment to active treatment is strictly between 0 and 1:* $0 < P(T = 1|\boldsymbol{X}) < 1$.

The last two assumptions imply that the treatment assignment is independent of the potential outcomes given the PS, i.e., $\{Y(0), Y(1)\} \perp T|\pi(\boldsymbol{X})$.

Let $F_t(y)$ be the unconditional cumulative distribution function (CDF) of the potential outcome $Y(t)$, defined as

$$F_t(y) = \int F_t(y|\boldsymbol{x}) dF(\boldsymbol{x}), \quad t = 0, 1,$$

where $F_t(y|\boldsymbol{x})$ is the conditional CDF of $Y(t)$ given covariates $\boldsymbol{X} = \boldsymbol{x}$ and $F(\boldsymbol{x})$ is the CDF of $\boldsymbol{X} = \boldsymbol{x}$. Following Assumptions 3 and 4, $F_t(y)$ is identifiable through $F_t(y|\boldsymbol{X})$. We make the additional assumption that for $\tau \in (0,1)$, $F_t(y)$ is twice differentiable and strictly increasing so that its inverse function is well defined. Let $q_t(\tau) = F_t^{-1}(\tau)$ denote the τth marginal quantile of $Y(t)$. The τth QTE (Doksum 1974; Lehmann and D'Abrera 1975) is defined as

$$\Delta_{\text{QTE}}(\tau) = q_1(\tau) - q_0(\tau),$$

which can be intuitively interpreted as the "horizontal distance" between the counterfactual distributions $F_1(y)$ and $F_0(y)$ in the target population.

Analogously, by assuming that $F_t(y|\boldsymbol{X})$ is twice differentiable and strictly increasing, the τth conditional quantile of $Y(t)$ given $\boldsymbol{X} = \boldsymbol{x}$, $q_t(\tau, \boldsymbol{x}) = F_t^{-1}(\tau|\boldsymbol{x})$, is also well defined. The τth conditional QTE (CQTE, Imbens and Wooldridge 2009) is defined as

$$\Delta_{\text{CQTE}}(\tau, \boldsymbol{x}) = q_1(\tau, \boldsymbol{x}) - q_0(\tau, \boldsymbol{x}).$$

In this chapter, we focus on the estimation of QTE as it is often the primary interest in causal inference. However, as we show in the next section, the CQTE can be estimated "for free" using the proposed method.

4.5 Methodology

4.5.1 Double balancing score

We propose to estimate the marginal quantiles $q_t(\tau)$ and the QTE by inverting a semi-parametric estimator of the counterfactual distribution, $\hat{F}_t(y)$. Although estimating the counterfactual distribution is a considerably more challenging problem than directly mod-

eling the quantile of interest, it avoids the necessity of fitting separate models for estimation of multiple quantiles. In addition, estimating the counterfactual distributions allows inference on not only quantiles but any functionals of the distributions.

We adopt a balancing score approach for estimation of the counterfactual distribution. Let $\boldsymbol{S}$ denote a balancing score which is a function of covariates satisfying the condition $T \perp \boldsymbol{X} \mid \boldsymbol{S}$. Our method begins by approximating the conditional counterfactual distribution given the covariates, $F_t(y|\boldsymbol{X})$, with an estimator of the conditional distribution of the observed outcome given the treatment and balancing score, $\hat{F}(y|T,\boldsymbol{S})$. A flexible estimator of $F_t(y)$ can then be obtained by marginalizing $\hat{F}(y|T=t,\boldsymbol{S})$ over the distribution of the balancing score, $F(\boldsymbol{S})$, i.e.,

$$\hat{F}_t(y) = \int \hat{F}(y|t,\boldsymbol{s}) d\, F(\boldsymbol{s}). \tag{4.1}$$

Most of the existing works set the balancing score to be either the full covariate vector (outcome-covariate regression) or the PS (outcome-PS regression). We shall argue that both approaches suffer from some drawbacks. When the full covariate vector, $\boldsymbol{X}$, is used as the balancing score, Equation (4.1) involves estimating a $(d+1)$-dimensional conditional distribution. In practice, d is often large as $\boldsymbol{X}$ has to include as many potential confounders as possible in order to satisfy the **SITA** assumption. In addition, covariates that are strong predictors of the outcome are often included in $\boldsymbol{X}$ to help explain the variability of the outcome more comprehensively. Thus estimation of $F(y|T,\boldsymbol{X})$ requires high-dimensional nonparametric regression which is an inherently difficult problem, especially when the outcome and covariates exhibit a rather complex relationship. Using the PS, $\pi(\boldsymbol{X})$, as the balancing score alleviates the curse of dimensionality by collapsing the full covariate vector to a probability and simplifies the estimation problem to a low-dimensional one. However, since the true PS is unknown in practice, a first-stage model is required to approximate $\pi(\boldsymbol{X})$ using data. This will introduce additional model uncertainty and inflate the variance of the estimated counterfactual distribution. Furthermore, conditioning on $\pi(\boldsymbol{X})$ may result in loss of information when $\pi(\boldsymbol{X})$ is not sufficient to explain the outcome and covariate relationship. To alleviate their individual drawbacks and combine their strength, we propose the use of an augmented score $\boldsymbol{S} = \{\pi(\boldsymbol{X}), \boldsymbol{X}\}^\top$. Since both components of $\boldsymbol{S}$ are balancing scores, $\boldsymbol{S}$ is a "double balancing score" (Hu et al. 2012). The semiparametric double balancing score estimator of the counterfactual distribution is still (4.1) but uses $\boldsymbol{S} = \{\pi(\boldsymbol{X}), \boldsymbol{X}\}^\top$ instead of only $\pi(\boldsymbol{X})$ or $\boldsymbol{X}$.

The proposed double balancing score estimation procedure has several advantages over

existing methods for counterfactual distribution estimation. Compared to outcome-PS regression approaches (Xu et al. 2018), the inclusion of the covariates makes $\boldsymbol{S}$ sufficient for explaining the outcome-covariate relationship, and therefore the resulting estimator may be more efficient. If the outcome-PS dependence relationship is complex but the outcome-covariate relationship is relatively simple, the proposed approach can benefit from the inclusion of covariates and estimate $F(y|T,\boldsymbol{X})$ more accurately. Compared to outcome-covariate regression approaches (Zhang et al. 2012; Chernozhukov et al. 2013), the inclusion of the PS incorporates signal from treatment assignment mechanism and utilizes the observed data fully. If the outcome-covariate relationship is complex but the outcome-PS dependence relationship is relatively simple, the proposed approach can benefit from the inclusion of PS and estimate $F(y|T,\boldsymbol{X})$ more accurately. The use of double balancing score is not the only way to aggregate information from both the covariates and the PS. Instead of including $\pi(\boldsymbol{X})$ as a regressor, one can use IPW to adjust for the bias of an initial conditional counterfactual distribution estimator (e.g., Kim et al. 2018; Kennedy et al. 2021). However, the advantage of incorporating $\pi(\boldsymbol{X})$ as one component of $\boldsymbol{S}$ is that associated drawbacks of IPW such as high variance due to subjects with extreme PS can be reduced.

4.5.2 Semiparametric counterfactual distribution estimation

As shown in (4.1), estimating $F_t(y)$ boils down to approximating the nonparametric conditional outcome CDF $F(y|T,\boldsymbol{S})$, or equivalently, the conditional outcome PDF $f(y|T,\boldsymbol{S})$. Without loss of generality, we assume that Y lies in the unit interval. This can be achieved by transforming Y from its original support using an appropriate monotone mapping. To allow a flexible model for the counterfactual density that can adapt to skewness, heavy-tailedness and multimodality, we extend the SPQR model by Xu and Reich (2021) to model the conditional distribution of outcomes given the treatment and balancing score. Let $\{M_k(y): 1 \le k \le K\}$ be K second-order M-spline basis functions with equally-spaced knots spanning $[0,1]$, and let $\{I_k(y): 1 \le k \le K\}$ be second-order I-spline basis functions with the same knots. We model the conditional PDF and CDF of Y respectively by

$$f(y|T,\boldsymbol{S}) = \sum_{k=1}^{K} \theta_k(T,\boldsymbol{S}) M_k(y) \quad \text{and} \quad F(y|T,\boldsymbol{S}) = \sum_{k=1}^{K} \theta_k(T,\boldsymbol{S}) I_k(y) \tag{4.2}$$

where the mixture weights $\theta_k(T, \boldsymbol{S})$ satisfy $\theta_k(T, \boldsymbol{S}) \geq 0$ and $\sum_{k=1}^{K} \theta_k(T, \boldsymbol{S}) = 1$ for all possible T and $\boldsymbol{X}$.

The weights are then modeled using feed-forward neural networks (NN) with L layers ($L-1$ hidden layers) and a softmax output activation. Let $\mathscr{W} = \{\boldsymbol{W}^{(1)}, ..., \boldsymbol{W}^{(L)}\}$ denote the set of weight matrices with $\boldsymbol{W}^{(l)} \in \mathbb{R}^{(V_l+1)\times V_{l-1}}$, where V_l is the number of units (excluding the intercept/bias node) in layer l and $V_0 = d+2$. Denote $\hat{\theta}_k(T, \boldsymbol{S}, \mathscr{W})$ as the NN estimator for $\theta_k(T, \boldsymbol{S})$ with modeling parameters $\mathscr{W}$, then $\hat{\theta}_k(T, \boldsymbol{S}, \mathscr{W})$ can be expressed hierarchically as

$$\begin{aligned}
&\hat{\theta}_k(T, \boldsymbol{S}, \mathscr{W}) = \text{softmax}\left\{z_k^{(L)}(T, \boldsymbol{S}, \mathscr{W})\right\}, \\
&z_k^{(l)}(T, \boldsymbol{S}, \mathscr{W}) = W_{k0}^{(l)} + \sum_{j=1}^{V_{l-1}} W_{kj}^{(l)} \phi\left\{z_j^{(l-1)}(T, \boldsymbol{S}, \mathscr{W})\right\} \quad \text{for } l = 2, ..., L, \\
&z_k^{(1)}(T, \boldsymbol{S}, \mathscr{W}) = W_{k0}^{(1)} + W_{k1}^{(1)} T + \sum_{j=1}^{d+1} W_{k(j+1)}^{(1)} S_j,
\end{aligned} \tag{4.3}$$

where $\text{softmax}(u_k) = e^{u_k} / \sum_{k=1}^{K} e^{u_k}$, $\phi(\cdot)$ is the hidden layer activation, and S_j is the jth component of $\boldsymbol{S}$. Throughout the chapter, we assume $\phi(\cdot)$ to be the hyperbolic tangent function, i.e., $\phi(u) = (e^{2u}-1)/(e^{2u}+1)$.

The NN is known to scale well with the dimension of the regressors, and thus the proposed estimator is particularly suitable for applications in which the observed covariates contain both confounders as well as predictors for only the outcome. The conditional distribution estimator combining (4.2) and (4.3) is extremely flexible. In fact, it has been shown that SPQR can approximate any smooth conditional distributions when $\boldsymbol{S} = \boldsymbol{X}$ (Xu and Reich 2021). The smoothness of regression splines also ensures that the quantiles and the causal effects are well defined.

4.5.3 Bayesian estimation of quantile causal effects

To allow uncertainty quantification of the estimated counterfactual distributions and quantile causal effects. We adopt a Bayesian framework for estimating the counterfactual distributions. A fully Bayesian approach using a PS requires posterior inference of the joint likelihood of the PS and outcome (Zigler et al. 2013), which may be complex especially when the PS is modeled nonparametrically. To simplify the problem, we adopt a sequential approach and estimate $\pi(\boldsymbol{X})$ and $F(y|T, \boldsymbol{S})$ separately in two stages. We use a fully

Bayesian approach to model and make inference on $\pi(\boldsymbol{X})$ using BART probit, $F(y|T,\boldsymbol{S})$ using Bayesian NN, and $F(\boldsymbol{S})$ using Bayesian bootstrap (Rubin 1981).

Propensity score

Since $\pi(\boldsymbol{X})$ is the conditional expectation of a binary variable, it is natural to use logistic regression to investigate the dependence relationship. However, parametric models are prone to misspecification when the relationship between the treatment and confounders is highly nonlinear. To ensure flexible estimation of the PS, we follow Xu et al. (2018) and model $\pi(\boldsymbol{X})$ using BART probit (Chipman et al. 2010), $\hat{\pi}(\boldsymbol{X}) = \Phi\left\{\sum_{j=1}^{m} \tilde{T}_j(\boldsymbol{X})\right\}$, where $\Phi(\cdot)$ is the standard normal CDF and $\tilde{T}(\cdot)$ is a regression tree model. Let $\left\{\hat{\pi}^{(k)}(\boldsymbol{X}_i),\ k = 1,...,N_\pi\right\}$ denote N_π posterior samples of $\hat{\pi}(\boldsymbol{X}_i)$, $i = 1,...,n$. We propagate uncertainty of the PS into the model of $F(y|T_i,\boldsymbol{S}_i)$ by treating $\left\{\boldsymbol{S}_i^{(k)},\ k = 1,...,N_\pi\right\}$ as informative priors for $\boldsymbol{S}_i$, where $\boldsymbol{S}_i^{(k)} = \left\{\hat{\pi}^{(k)}(\boldsymbol{X}_i), \boldsymbol{X}_i\right\}^\top$. We then estimate $F(y|T_i,\boldsymbol{S}_i^{(k)})$ using (4.3) for each k.

Priors over NN weights

To complete the Bayesian formulation for the conditional distribution estimator, we must assign priors to the NN weights in SPQR. The uncertainty of $f(y|T,\boldsymbol{S})$ and $F(y|T,\boldsymbol{S})$ can then be characterized by the posterior distributions of $\hat{\theta}_k(T,\boldsymbol{S},\mathscr{W})$. Zero-mean Gaussian distributions are widely used as priors for Bayesian NN weights owning to their *weight-decay* property. They have been explored both in classic works on shallow NNs (Neal 1996; MacKay 1992) as well as more recent works on deep NNs (Matthews et al. 2018; Fortuin et al. 2021). The performance of Gaussian priors depends heavily on the variance hyper-prior structure which allows one to encode problem-specific beliefs as well as general properties about weights. An isotropic Gaussian prior that assigns all weights a common variance, although a convenient choice, can in fact lead to inflated predictive uncertainties (Neal 1996).

Gaussian scale mixture (GSM) priors have been shown to be effective for Bayesian inference of large NNs (e.g., Cui et al. 2021). For each layer $l \in \{1,...,L\}$, the GSM prior on $W_{kj}^{(l)}$ can be expressed hierarchically as

$$W_{kj}^{(l)}|\sigma^{(l)},\lambda_j^{(l)} \sim \mathscr{N}\left(0,\sigma^{(l)2}\lambda_j^{(l)2}\right),\ p(\lambda_j^{(l)}) \sim p(\lambda_j^{(l)};\gamma_\lambda) \quad \text{for} \quad j \geq 0, \tag{4.4}$$

where $\sigma^{(l)}$ is the layer-wise global scale shared by all weights in layer l, which can either be

set to a constant value or estimated using non-informative priors, and $\lambda_j^{(l)}$ is a unit-wise local scale with hyper-prior $p(\lambda_j^{(l)};\gamma_\lambda)$. Although many more advanced settings are equally applicable (e.g., Ghosh et al. 2019), we use inverse-Gamma distributions as hyper-priors for both the global variance $\sigma^{(l)2}$ and the local variance $\lambda_j^{(l)2}$, i.e.,

$$p(\sigma^{(l)2};\gamma_\sigma) = \textit{Inv-}\mathcal{G}\textit{amma}(a_\sigma, b_\sigma) \quad \text{and} \quad p(\lambda_j^{(l)2};\gamma_\lambda) = \textit{Inv-}\mathcal{G}\textit{amma}(a_\lambda, b_\lambda),$$

as it simplifies the sampling algorithm greatly. By assigning all outgoing weights $W_{kj}^{(l)}$ from node j in layer l a common scale parameter, the GSM prior achieves *Automatic Relevance Determination* (ARD) (MacKay 1992) which allows weights that are associated with relevant features to be large and forces weights that are associated with irrelevant features to be small. The ARD property is especially appealing to the current context since $f(y|T,\boldsymbol{S})$ is often high-dimensional to satisfy **SITA**, and high-dimensional conditional distributions often have a *sparse* structure (Izbicki and Lee 2016). In addition, the hierarchical structure of GSM prior allows characterization of heavy-tailed weight distributions which are often observed in deep NNs (Fortuin et al. 2021).

To sample the parameters $W_{ij}^{(l)}$, $\sigma^{(l)}$ and $\lambda_j^{(l)}$ in (4.4), we use Markov chain Monte Carlo (MCMC) methods to approximate their posterior distributions. Specifically, we follow the strategy of Neal (1996) and use a block-updating scheme that combines the strength of two MCMC algorithms. The posterior distribution of the weight parameters $W_{kj}^{(l)}$ is high-dimensional and has a complex geometry

$$p(\mathscr{W};\sigma^{(l)},\lambda_j^{(l)}) \propto \prod_{l=1}^{L}\prod_{k=1}^{K}\prod_{j=0}^{V_{l-1}} \mathcal{N}\left(W_{kj}^{(l)}|0,\sigma^{(l)2}\lambda_j^{(l)2}\right) \times \prod_{i=1}^{n} f(y_i|T_i,\boldsymbol{S}_i,\mathscr{W}),$$

therefore it is approximated using the no-U-turn sampler (NUTS, Hoffman and Gelman 2014). The prior distributions of the variance hyperparameters $\sigma^{(l)2}$ and $\lambda_j^{(l)2}$ are conjugate, therefore their full conditional distributions can be derived analytically as

$$\sigma^{(l)2}|W_{kj}^{(l)},\lambda_j^{(l)} \sim \textit{Inv-}\mathcal{G}\textit{amma}\left(a_\sigma + \frac{V^{(l+1)}(V^{(l)}+1)}{2}, b_\sigma + \frac{\sum_{k=1}^{K}\sum_{j=0}^{V_{l-1}}\left(W_{kj}^{(l)}/\lambda_j^{(l)}\right)^2}{2}\right)$$

$$\lambda_j^{(l)2}|W_{kj}^{(l)},\sigma^{(l)} \sim \textit{Inv-}\mathcal{G}\textit{amma}\left(a_\lambda + \frac{V^{(l+1)}}{2}, b_\lambda + \frac{\sum_{k=1}^{K}\left(W_{kj}^{(l)}/\sigma^{(l)}\right)^2}{2}\right)$$

and their posterior distributions can be approximated using Gibbs sampling (Geman and Geman 1984).

Bayesian bootstrap

Finally, to estimate the unconditional counterfactual distribution, we need to marginalize the estimated conditional distribution of outcomes given the treatment and the balancing score over the population distribution of the balancing score. The population distribution of the covariates (and the balancing score) is typically not known in observational studies. Under the assumption that the observed group of individuals is a simple random sample from the target population, a reasonable estimate of the covariate distribution is the empirical distribution $H_n(\boldsymbol{X}) = \frac{1}{n}\sum_{i=1}^{n} \delta_{\boldsymbol{X}_i}$ where $\delta_{\boldsymbol{X}_i}$ is a degenerate distribution at $\boldsymbol{X}_i$. Similarly, for each posterior sample of the balancing score, $\boldsymbol{S}^{(k)}$, we can estimate its distribution by $H_n^{(k)}(\boldsymbol{S}) = \frac{1}{n}\sum_{i=1}^{n} \delta_{\boldsymbol{S}_i^{(k)}}$. To incorporate uncertainty of this distribution, we follow the suggestion of Xu et al. (2018) and use the Bayesian bootstrap (Rubin 1981). Consider the weighted empirical distribution model $H_n^{(k)}(\boldsymbol{S}, \boldsymbol{u}) = \sum_{i=1}^{n} u_i \delta_{\boldsymbol{S}_i^{(k)}}$ where the sampling weights u_i are unknown, non-negative, and sum to one. We give the weights non-informative priors $p(\boldsymbol{u}) \propto \prod_{i=1}^{n} u_i^{-1}$. The uncertainty of the balancing score distribution can then be characterized by the posterior distribution of $\boldsymbol{u}$ which follows $\mathscr{D}irichlet(1, ..., 1)$.

The steps for estimating the counterfactual distribution and QTE are summarized in Algorithm 1. After obtaining the posterior samples: $f_0^{(jl)}(y)$, $f_1^{(jl)}(y)$ and $\Delta_{\text{QTE}}^{(jl)}(\tau)$, for $j = 1, ..., N_\pi$ and $l = 1, ..., N_{\mathscr{W}}$. We can compute the estimated counterfactual densities and QTE by averaging the samples

$$\hat{f}_t(y) = \frac{1}{N_\pi}\frac{1}{N_{\mathscr{W}}}\sum_{j=1}^{N_\pi}\sum_{l=1}^{N_{\mathscr{W}}} f_t^{(jl)}(y) \quad \text{for} \quad t = 0, 1$$

$$\widehat{\Delta}_{\text{QTE}}(\tau) = \frac{1}{N_\pi}\frac{1}{N_{\mathscr{W}}}\sum_{j=1}^{N_\pi}\sum_{l=1}^{N_{\mathscr{W}}} \Delta_{\text{QTE}}^{(jl)}(\tau)$$

and compute credible intervals (CIs) using relevant percentiles. As mentioned in Section 4.4, the proposed method can simultaneously estimate the CQTE together with QTE. Let $q_t^{(jl)}(\tau, \boldsymbol{s}^{(j)})$ denote the samples of the conditional quantile given $\boldsymbol{X} = \boldsymbol{x}$ obtained by inverting the samples of conditional outcome distribution in step 8, i.e.,

$$F^{(jl)}(q_t^{(jl)}(\tau, \boldsymbol{s}^{(j)}) | T = t, \boldsymbol{s}^{(j)}) = \tau,$$

and construct the posterior samples of CQTE as $\Delta^{(jl)}_{\text{CQTE}}(\tau, \boldsymbol{x}) = q_t^{(jl)}(\tau, \boldsymbol{s}^{(j)}) - q_t^{(jl)}(\tau, \boldsymbol{s}^{(j)})$. We can analogously compute the estimated CQTE, $\widehat{\Delta}^{(jl)}_{\text{CQTE}}(\tau, \boldsymbol{x})$, by averaging the samples and compute CIs using relevant percentiles.

4.6 Simulation

We examine the performance of the proposed approach in estimating counterfactual densities and QTE using four simulations. The details for each simulation are provided as follows. (1) Simulation 1 explores the effect of adjusting for individual covariates in addition to the PS in presence of nonconfounding covariates. (2) Simulation 2 explores the effect of adjusting for the PS in addition to the covariates when the outcome-covariate relationship is complex but the outcome-PS relationship is simple. (3) Simulation 3 explores the effect of adjusting for the covariates in addition to the PS when the outcome-PS relationship is complex but the outcome-covariate relationship is simple. (4) Simulation 4 investigates the performance of the proposed approach when the conterfactual distributions are strongly non-Gaussian. For all four simulations, the proposed approach, which we call SPQR-DS (double score), is compared with two counterfactual distribution estimators: the DPM-BART estimator proposed by Xu et al. (2018) and the truncated series (TS) estimator proposed by Kennedy et al. (2021). The DPM-BART estimator uses outcome-PS regression. First, the PS is modeled flexibly using BART probit (Chipman et al. 2010). For each treatment group, the distribution of potential outcomes conditional on the estimated PS is estimated using a Dirichlet process mixture (DPM) of normals and finally marginalized over the population distribution of the covariates. Similar to the proposed approach, the DPM estimator incorporate uncertainty of the PS through its posterior samples and the uncertainty of the covariate distribution through Bayesian bootstrap. The TS estimator uses a doubly robust like approach. First, the distribution of the potential outcomes conditional on the full covariate vector is estimated using the kernel-smoothed approach of Kim et al. (2018). An intial estimate of the counterfactual density is then obtained by projecting the nonparametric kernel estimator to a truncated cosine series. Finally, IPW is combined with this initial estimate for bias correction. In addition to the three main approaches described above, we also compared DPM for modeling the entire distribution of the outcome given covariates (DPM-X), and the proposed estimator that uses either only the individual covariates (SPQR-X) or only the PS (SQRT-BART). We did not compare with DPM that uses double balancing score because

Algorithm 1 Bayesian semiparametric estimation of counterfactual densities and QTE

Input: Observed data $(\boldsymbol{X}_i, T_i, Y_i)$

Output: Posterior samples of $f_0(y)$, $f_1(y)$ and $\Delta_{\text{QTE}}(\tau)$

Sample $\hat{\pi}^{(j)}(\boldsymbol{X}_i)$, $j = 1, ..., N_\pi$, from BART probit

for $j = 1, ..., N_\pi$ **do**

  $\boldsymbol{S}_i^{(j)} \leftarrow \left\{\hat{\pi}^{(j)}(\boldsymbol{X}_i), \boldsymbol{X}_i\right\}^\top$

  Sample $\mathscr{W}^{(jl)}$, $l = 1, ..., N_{\mathscr{W}}$, from Bayesian NN ▷ Equation (4.4)

  for $l = 1, ..., N_{\mathscr{W}}$ **do**

    Sample $u_i^{(jl)}$ from $\mathscr{D}irichlet(1, ..., 1)$; $H_n^{(jl)}(\boldsymbol{S}, \boldsymbol{u}) = \sum_{i=1}^n u_i^{(jl)} \delta_{\boldsymbol{S}_i^{(j)}}$

    for $t = 0, 1$ **do**

      Compute ▷ Equation (4.2)

$$F^{(jl)}(y|T=t, \boldsymbol{S}_i^{(j)}) = \sum_{k=1}^{K} \hat{\theta}_k\left(t, \boldsymbol{S}_i^{(j)}, \mathscr{W}^{(jl)}\right) I_k(y)$$

$$f^{(jl)}(y|T=t, \boldsymbol{S}_i^{(j)}) = \sum_{k=1}^{K} \hat{\theta}_k\left(t, \boldsymbol{S}_i^{(j)}, \mathscr{W}^{(jl)}\right) M_k(y)$$

      Compute ▷ Equation (4.1)

$$\begin{aligned} F_t^{(jl)}(y) &= \int F^{(jl)}(\tilde{y}_g|T=t, \boldsymbol{s}^{(j)}) d H_n^{(jl)}(\boldsymbol{s}^{(j)}) \\ &= \sum_{i=1}^{n} u_i^{(jl)} F^{(jl)}(\tilde{y}_g|T=t, \boldsymbol{S}_i^{(j)}) \end{aligned}$$

$$f_t^{(jl)}(y) = \sum_{i=1}^{n} u_i^{(jl)} f^{(jl)}(\tilde{y}_g|T=t, \boldsymbol{S}_i^{(j)})$$

      $q_t^{(jl)}(\tau) \leftarrow y$ s.t. $F_t(y) = \tau$

    end for

    Compute $\Delta_{\text{QTE}}^{(jl)}(\tau) = q_1^{(jl)}(\tau) - q_0^{(jl)}(\tau)$

  end for

end for

return $f_0^{(jl)}(y)$, $f_1^{(jl)}(y)$ and $\Delta_{\text{QTE}}^{(jl)}(\tau)$, $j = 1, ..., N_\pi$ and $l = 1, ..., N_{\mathscr{W}}$

the perfect multicollinearity renders DPM numerically unstable.

To estimate the PS, we sample the posterior distribution of the parameters in the BART probit model using the R package **BayesTree** with default priors (Chipman et al. 2010). We run the MCMC for 1000 iterations and save every 100th iteration after discarding 500 iterations as burn-in. To estimate the conditional distribution using SPQR, we use the R package **SPQR** with the priors described in Section 4.5.3. We set $a_\sigma = b_\sigma = a_\lambda = b_\lambda = 0.01$ to give the variance parameters uninformative priors. We model the mixture weights using NNs with one hidden layer and select the number of basis functions and hidden neurons from $K = \{8, 10, 12\}$, $V_1 = \{5, 8, 10\}$ using WAIC (Watanabe 2013). We run the MCMC for 3000 iterations and save every 10th iteration after discarding 1000 iterations as burn-in. To estimate the conditional distribution using DPM of normals, we use the R code in Xu et al. (2018). Following the original authors' setup, we run the MCMC for 900 iterations save every 2nd iteration after discarding 500 iterations as burn-in. For TS, we use the R package **npcasual** to fit the model and its built-in cross-validation (CV) routine for basis selection. For each level of treatment, we consider up to 10 basis terms and choose the best model using 5-fold CV. During our experiment, we found that the algorithm used to fit the TS model is sometimes numerically unstable due to multiple uses of numerical integration. In cases when CV fails, we use the default setting of five basis functions for both treatment groups.

Each simulation design generates covariates $\boldsymbol{X}$, then binary treatments $T|\boldsymbol{X}$, and finally potential outcomes $Y(0)$ and $Y(1)$ for each counterfactual regime. The observed response is then $Y = TY(1) + (1-T)Y(0)$. We generate 100 replicated datasets with sample size $n = 500$ for each design. We compare the estimated density of potential outcomes from the different approaches. In particular, we use the integrated squared error (ISE) to quantify the difference between an estimated density and its ground truth. We estimate the ISE by

$$\begin{aligned}\text{ISE}(\hat{f}, f) &\equiv \int \left[\hat{f}(y) - f(y)\right]^2 dy \\ &\approx (g_1 - g_0) \sum_{i=1}^{n_{\text{grid}}-1} \left[\hat{f}(g_i) - f(g_i)\right]^2 + \frac{\left[\hat{f}(g_0) - f(g_0)\right]^2 + \left[\hat{f}(g_{n_{\text{grid}}}) - f(g_{n_{\text{grid}}})\right]^2}{2}\end{aligned}$$

where $\hat{f}(y)$ is the estimated density, $f(y)$ is the true density, g_i are equidistant grid points and $n_{\text{grid}} = 200$ is used in the simulations. We calculate QTE for 19 quantiles ($\tau = 0.05, 0.10, \ldots, 0.95$). We compare all the different approaches in terms of τ-specific root mean squared

error (RMSE) and average absolute bias (AAB)

$$\text{RMSE}(\tau) = \frac{1}{100}\sum_{i=1}^{100}\left[\widehat{\Delta}^{(i)}_{\text{QTE}}(\tau) - \Delta_{\text{QTE}}(\tau)\right],$$
$$\text{AAB} = \frac{1}{19}\sum_{i=1}^{19}\left|\widehat{\Delta}_{\text{QTE}}(\tau_i) - \Delta_{\text{QTE}}(\tau_i)\right|,$$

where $\widehat{\Delta}^{(i)}_{\text{QTE}}(\tau)$ denote the estimated τ-QTE from the ith replicated data set.

4.6.1 Simulation 1

Each subject is associated with 5 continuous covariates of which J are confounders and $5-J$ are nonconfounding covariates. The true PS model is a logistic regression model that includes only main effects of the confounders. One potential outcome is a mixture of normal models and the other has a skewed distribution for the error term. The exact form of the true model is

$$\begin{aligned}
X_j &\sim \mathscr{U}(-2,2) \quad j = 1,\ldots,5\\
T|\boldsymbol{X} &\sim \mathscr{B}ern\left(\text{expit}\left(\frac{4}{J}\sum_{j=1}^{J} X_j\right)\right)\\
Y(0)|\boldsymbol{X} &= -2.3 + Z_1 + Z_1^2 + \epsilon\\
\epsilon &\sim 0.75\mathscr{N}^{+}(0,0.9^2) + 0.25\mathscr{N}^{-}(0,0.3^2)\\
Y(1)|\boldsymbol{X} &\sim 0.7\mathscr{N}\left(-2.5+5Z_2, 0.35^2\right) + 0.3\mathscr{N}\left(2.5-5Z_2, 0.35^2\right)
\end{aligned}$$

where $Z_k = \text{expit}(0.8\sum_{j=1}^{5} X_j + 0.1\sum_{j=1}^{5}|X_j|^k)$. We experiment with two settings of J: $J=0$ such that there is no confounding and the data represent observations from a randomized trial, and $J=2$ such that there is strong confounding. In both cases, nonconfounding covariates contribute to variability of outcome residual after adjusting for PS.

In the case of $J=0$, Fig. 4.1 displays the true and estimated counterfactual PDFs and CDFs from the 100 replicates for SPQR-DS, DPM-BART and TS. The results show that all three approaches correctly estimate the bimodal and skewed distributions of the potential outcomes. However, compared to SPQR-DS and DPM-BART, the TS model has significantly larger variance in the tails of the distribution. Fig. 4.2(a) displays the boxplots of ISE for all approaches. The results show that despite only using the PS, the DPM-BART approach

Table 4.1: Simulation 1. AAB and τ-specific RMSE of QTE for all approaches. AAB is calculated using all 19 quantiles, and standard deviation is given in parentheses.

	RMSE of $\widehat{\Delta}_{\text{QTE}}(\tau)$					
τ	SPQR-DS	SPQR-BART	SPQR-X	DPM-BART	DPM-X	TS
$J=0$						
0.1	0.10	0.09	0.11	0.10	0.12	0.11
0.25	0.16	0.16	0.16	0.16	0.21	0.14
0.5	0.21	0.22	0.21	0.20	0.15	0.21
0.75	0.12	0.15	0.12	0.12	0.23	0.15
0.9	0.11	0.13	0.13	0.11	0.12	0.11
AAB	**0.12 (0.06)**	0.13 (0.07)	**0.12 (0.06)**	**0.12 (0.06)**	0.16 (0.04)	**0.12 (0.05)**
$J=2$						
0.1	0.15	0.16	0.17	0.24	0.19	0.24
0.25	0.21	0.24	0.21	0.44	0.28	0.34
0.5	0.24	0.30	0.30	0.28	0.24	0.36
0.75	0.17	0.20	0.23	0.16	0.29	0.21
0.9	0.19	0.22	0.21	0.21	0.17	0.20
AAB	**0.17 (0.06)**	0.19 (0.10)	0.19 (0.08)	0.23 (0.10)	0.20 (0.08)	0.23 (0.10)

Note: best-performing models with the smallest average AAB are in bold fonts.

yields the best performance in density estimation. One possible explanation might be that when there is no confounding, $f_t(y)$ does not require identification through $f_t(y|\boldsymbol{X})$ Therefore the normal outcome distributions can be easily approximated by DPM since it is the correct model. On the other hand, directly estimating the full conditional using DPM yields poor performance for the control density, suggesting that DPM may not scale well with the dimension of the conditioning variables. The proposed approach seems to benefit from the inclusion of both the PS and the full covariate, yielding smaller variance than its two sub-models. The same plots in the case of $J=2$ are shown in Fig. 4.3 and 4.2(b). The results show that even in presence of strong confounding, the proposed approach still yields competitive accuracy for counterfactual distributions. The DPM-BART approach yields biased estimates for the treatment distribution. The TS approach is generally accurate, but is unstable and yields more outlying estimates. Table 4.1 displays the RMSE of QTEs at five selected quantile levels and AAB based on all 19 quantile levels for all approaches. The results show that the proposed model yields the best QTE estimation in both cases. The use of double-balancing score for residual adjustment is particularly advantageous in presence of strong confounding, reducing the bias and improving the efficiency of the estimator.

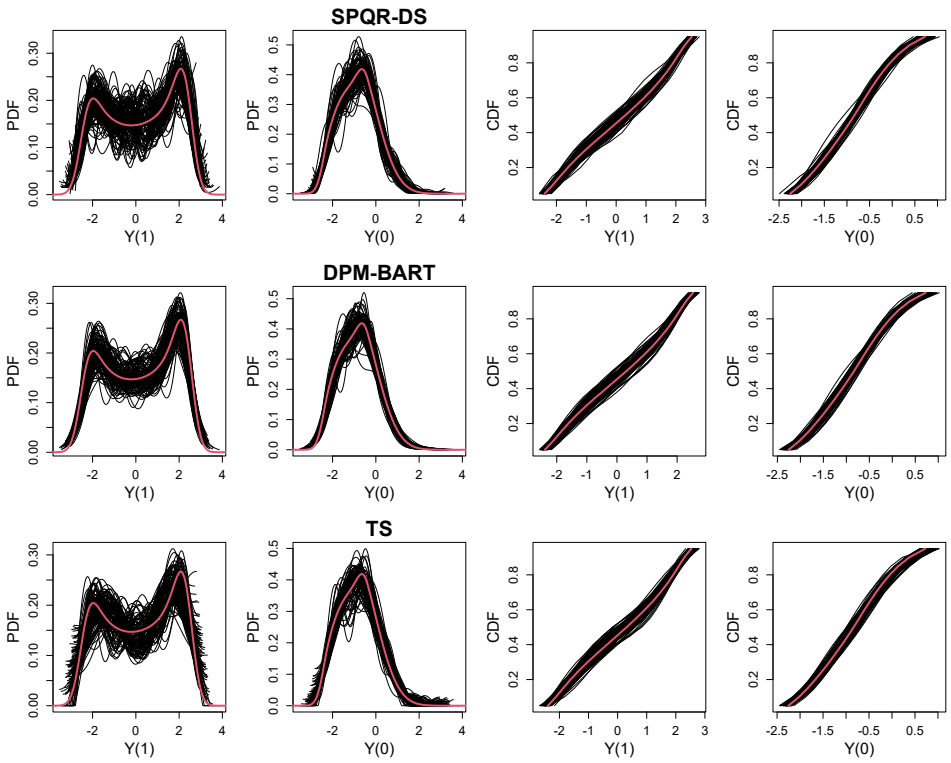


Figure 4.1: Simulation 1, $J = 0$: Estimated PDFs and CDFs (black lines) of potential outcomes compared to the ground truth (red line), for 100 replicates for the SPQR-DS approach, the DPM-BART approach, and the TS approach.

4.6.2 Simulation 2

Each subject is associated with 6 continuous and 6 binary confounders. The true PS model is a logistic regression model that includes interactions between the confounders. Both potential outcomes follow mixture of normal models with the normal distributions depending on functions of the PS. In addition, the normal mixture model under control has weights depending on the PS. The exact form of the true model is given in Appendix B.1.

Fig. 4.4 displays the true and estimated counterfactual PDFs and CDFs from the 100 replicates for SPQR-DS, DPM-BART and TS. The results show that all three approaches correctly estimate the counterfactual distributions. Fig. 4.2(c) displays the boxplots of ISE for all approaches except DPM since the algorithm failed on all 100 replicates. This suggests that DPM is generally not suitable for high-dimensional conditional distribution regression. The results show that all approaches that incorporate information of PS perform better

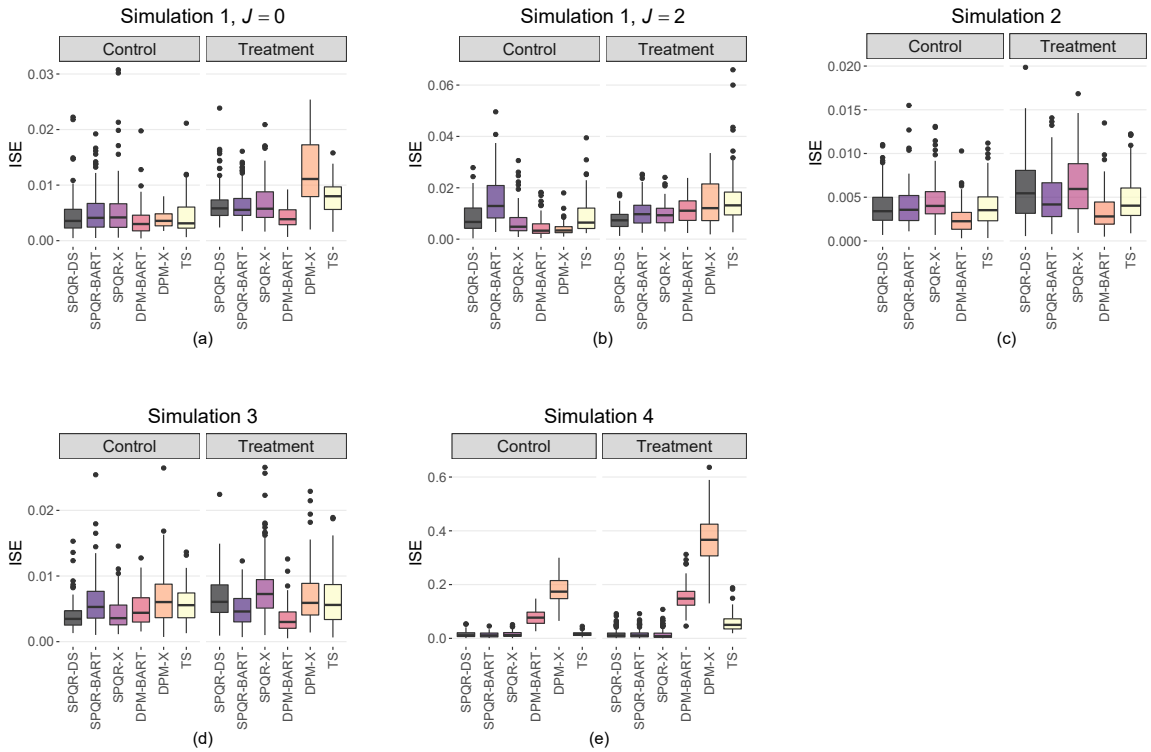


Figure 4.2: Simulation results. ISE of estimated counterfactual densities for 100 replicates for all approaches.

than those who do not (in this case only SPQR-X). This is expected since the outcome-PS dependence relationship has a simple form. The DPM-BART approach is again the top-performer in terms of ISE, possibly due to the fact that the true counterfactual distributions are mixtures of normals. Table 4.2 displays the RMSE and AAB of QTEs for all approaches. The results show that the proposed approach achieves similar performance as DPM-BART in QTE estimation. Compared to SPQR-BART and SPQR-X, the proposed approach reduces the bias of almost all estimated QTEs, giving clear evidence of the advantage of double adjustment. The overall lower bias of the SPQR-DS and DPM-BART estimators compared to the TS estimator also suggests that the information of the treatment assignment mechanism is better utilized through PS adjustment than IPW in this specific setting.

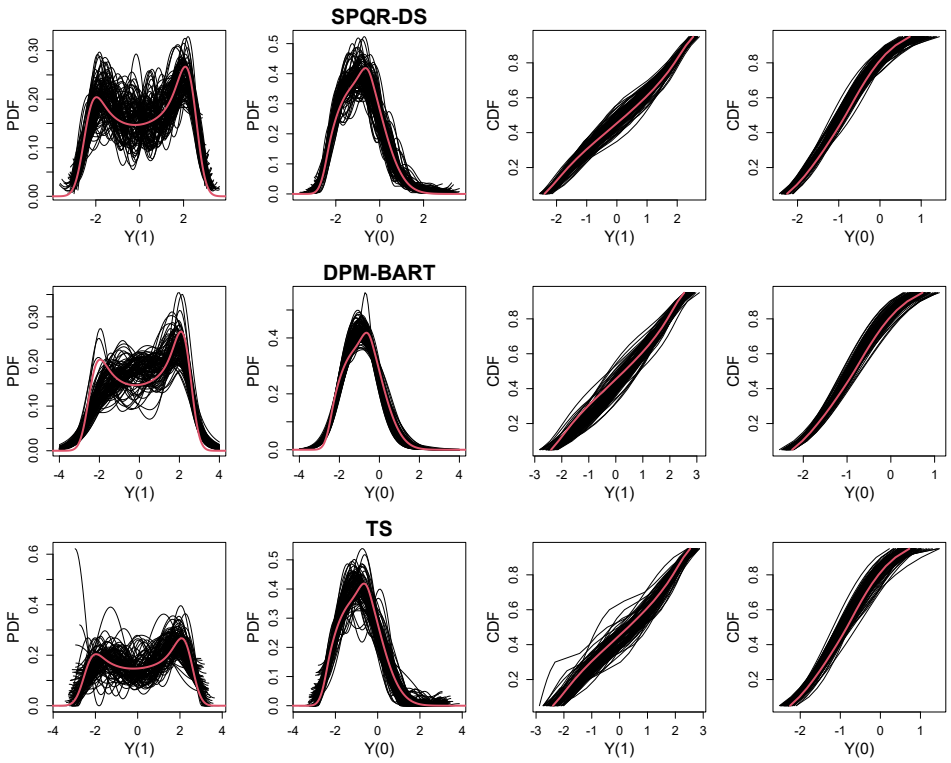


Figure 4.3: Simulation 1, $J = 2$: Estimated PDFs and CDFs (black lines) of potential outcomes compared to the ground truth (red line), for 100 replicates for the SPQR-DS approach, the DPM-BART approach, and the TS approach.

4.6.3 Simulation 3

Each subject is associated with 4 continuous confounders. The true PS model is a NN model with one hidden layer and five hidden neurons. One potential outcome follows a normal distribution and the other follows a skew normal distribution. The exact form of the true model is given in Appendix B.2.

Fig. 4.5 displays the true and estimated counterfactual PDFs and CDFs from the 100 replicates for SPQR-DS, DPM-BART and TS. The results show that all three approaches correctly estimate the counterfactual distributions. Fig. 4.2(d) displays the boxplots of ISE for all approaches. The proposed approach performs the best in estimating the control density, whereas the DPM-BART approaches performs the best in estimating the treatment density. The results show that even though the outcome-PS dependence relationship is quite complex, the BART probit model is flexible enough to approximate it reasonably well.

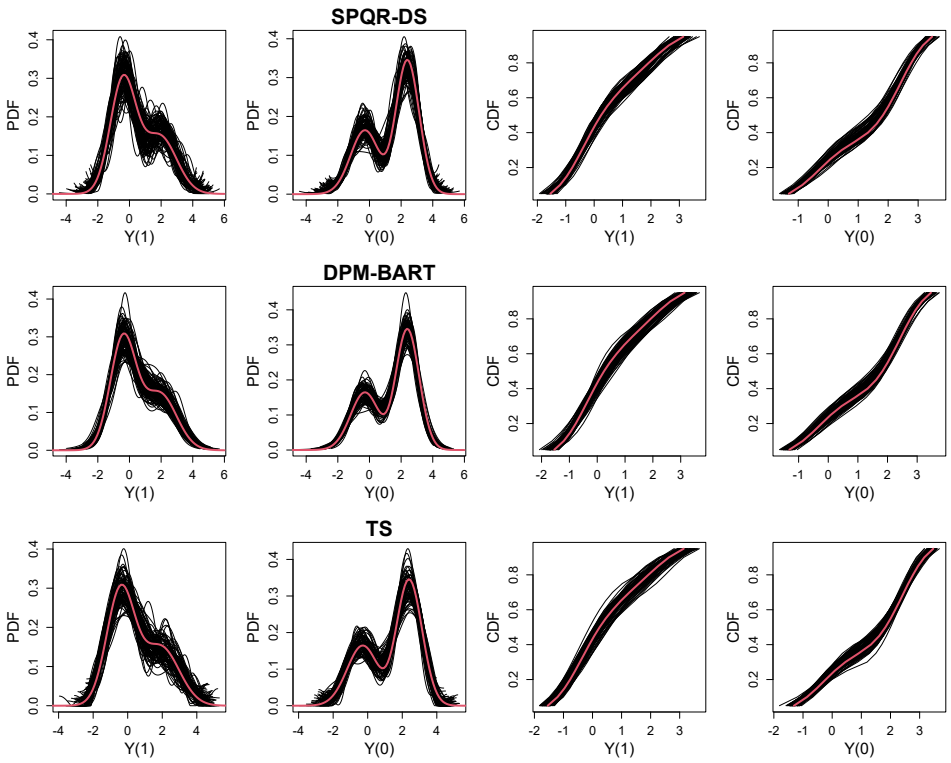


Figure 4.4: Simulation 2: Estimated (black lines) and true (red line) distributions of potential outcomes for 100 replicates.

Table 2 displays the RMSE and AAB of QTEs for all approaches. The results show that the proposed approach achieves the best performance in QTE estimation. Comparison with SPQR-BART and DPM-BART shows that the proposed approach benefits from additionally adjusting for the individual covariates which help explain the variability of outcome residual after adjusting for the PS. Comparison with SPQR-X and DPM-X shows that the inclusion of the PS improves the estimation of the conditional distribution in presence of strong confounding.

4.6.4 Simulation 4

Given that the true counterfactual distributions in Simulation 1 to 3 are all based on normal distributions or skew normal distributions, it is not surprising to see that the DPM-BART approach consistently yields excellent performance in estimating the counterfactual distributions. We show in this simulation example that the use of a parametric mixing distribution

Table 4.2: Simulation 2 – 4. AAB and τ-specific RMSE of QTE for all approaches. AAB is calculated using all 19 quantiles, and standard deviation is given in parentheses.

	RMSE of $\widehat{\Delta}_{\text{QTE}}(\tau)$					
τ	SPQR-DS	SPQR-BART	SPQR-X	DPM-BART	DPM-X	TS
Simulation 2						
0.1	0.14	0.13	0.15	0.15	–	0.16
0.25	0.16	0.16	0.19	0.17	–	0.17
0.5	0.20	0.23	0.23	0.21	–	0.2
0.75	0.21	0.26	0.24	0.20	–	0.21
0.9	0.18	0.20	0.19	0.19	–	0.23
AAB	**0.15 (0.07)**	0.16 (0.09)	0.17 (0.09)	**0.15 (0.07)**	–	0.16 (0.07)
Simulation 3						
0.1	0.08	0.08	0.08	0.08	0.08	0.10
0.25	0.09	0.12	0.09	0.10	0.11	0.11
0.5	0.12	0.17	0.12	0.16	0.17	0.17
0.75	0.16	0.19	0.17	0.17	0.16	0.21
0.9	0.13	0.14	0.13	0.13	0.18	0.13
AAB	**0.09 (0.04)**	0.12 (0.05)	0.10 (0.04)	0.11 (0.04)	0.12 (0.04)	0.12 (0.05)
Simulation 4						
0.1	0.01	0.01	0.01	0.01	0.02	0.01
0.25	0.02	0.02	0.02	0.03	0.03	0.02
0.5	0.03	0.03	0.04	0.04	0.06	0.03
0.75	0.06	0.06	0.06	0.06	0.06	0.06
0.9	0.12	0.11	0.12	0.11	0.13	0.17
AAB	**0.04 (0.02)**	**0.04 (0.02)**	**0.04 (0.02)**	**0.04 (0.02)**	0.05 (0.02)	**0.04 (0.02)**

Note: best-performing models with the smallest average AAB are in bold fonts.

can become restrictive in certain cases, whereas the proposed approach is more flexible with its use of spline basis.

For simplicity, we set the true $\pi(\boldsymbol{X})$ to be 0.5 so that the data represent a randomized experiment setup. Each subject is associated with 5 continuous covariates that have no effect on the outcome distribution. The potential outcomes have exponential distributions

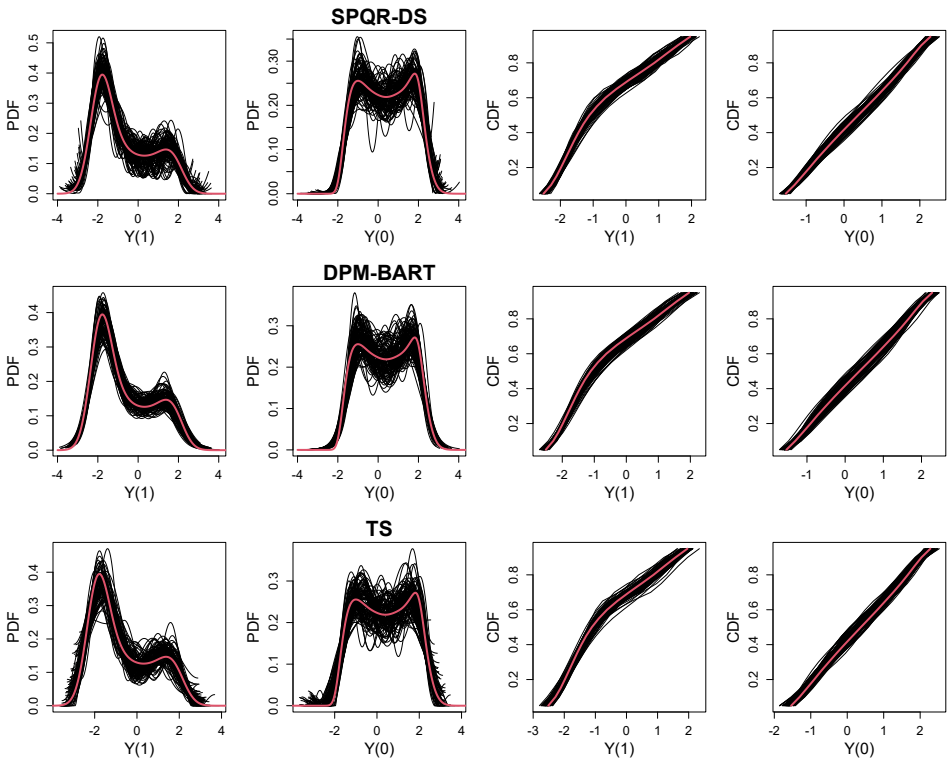


Figure 4.5: Simulation 3: Estimated (black lines) and true (red line) distributions of potential outcomes for 100 replicates.

with different rate parameters. The exact form of the true model is

$$\begin{aligned} X_j &\sim \mathcal{U}(-2,2) \quad j = 1,...,5 \\ T|\boldsymbol{X} &\sim \mathcal{B}ern(0.5) \\ Y(0)|\boldsymbol{X} &\sim \mathcal{E}xp(2) \\ Y(1)|\boldsymbol{X} &\sim \mathcal{E}xp(4). \end{aligned}$$

The exponential distribution is difficult to approximate with a normal mixture since it does not have a well defined mode.

Fig. 4.6 displays the true and estimated counterfactual PDFs and CDFs from the 100 replicates for SPQR-DS, DPM-BART and TS. The results show that the proposed approach and the TS approach correctly capture the monotonic nature of the underlying densities, whereas the DPM-BART gives a misleading description of the densities as it tries to fit a unimodal distribution to the data. However, the DPM-BART estimates the CDFs correctly.

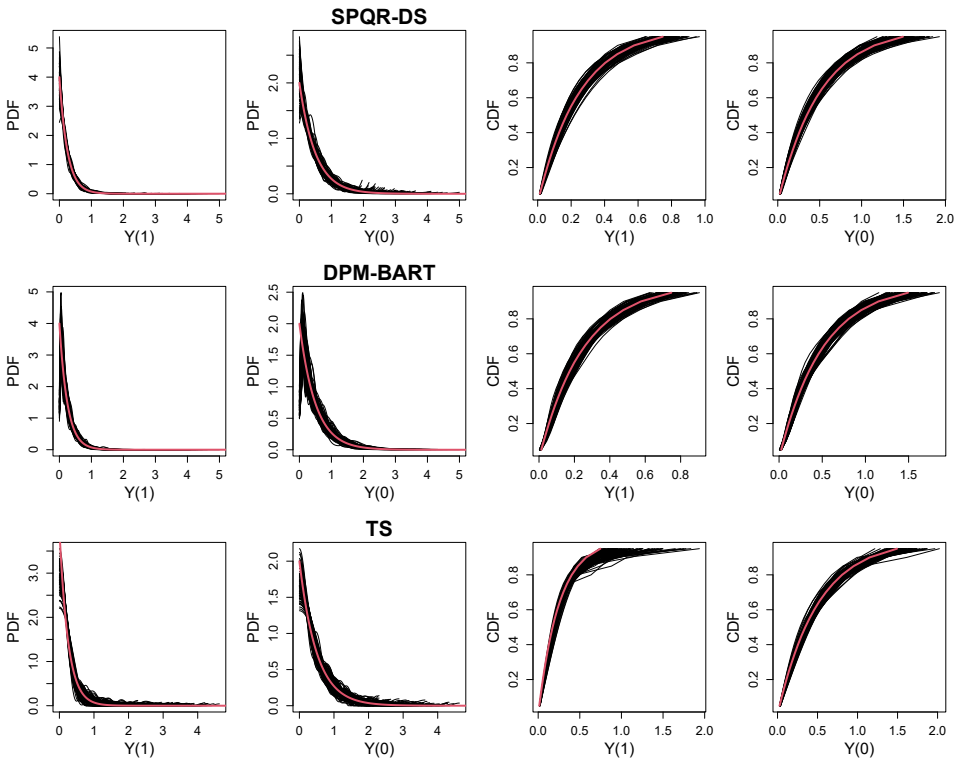


Figure 4.6: Simulation 4: Estimated (black lines) and true (red line) distributions of potential outcomes for 100 replicates.

This shows that the counterfactual PDFs indeed provide more nuanced information than the CDFs and are more difficult to estimate accurately. Thus, a model that can accurately estimate both the counterfactual PDFs and CDFs is crucial for correct inference on the counterfactual distributions. Fig. 4.2(e) displays the boxplots of ISE for all approaches. The results show that the SPQR-based approaches yield smallest ISE, whereas the DPM-based approaches yield the largest ISE. Table 2 displays the RMSE and AAB of QTEs for all approaches. The results show that all approaches yield similar overall performance on QTE estimation. However, compared to the other approaches, the TS approach yields significantly worse performance on estimating upper-tail QTEs. This again shows that the TS approach has poor boundary properties due to the oscillating nature of cosine series.

4.7 Data application

In this section, we apply our proposed method to estimate the causal effect of maternal smoking on birth weight distribution. In the literature, many studies have shown that low birth weight is associated with increased risk of health problems after birth and long-term economic cost due to medication, and that maternal smoking is one of the major modifiable risk factor for low birth weight (Abrevaya 2001; Ngwira and Stanley 2015). Consequently, there has been a great deal of interest in studying the causal effect of maternal smoking on infant birth weight (e.g., Abrevaya et al. 2015; Huang and Yang 2020; Xie et al. 2020; Zhou et al. 2021). We adopt a data set based on records between 1988 and 2002 by the North Carolina Center Health Services. This data set was analyzed by Abrevaya et al. (2015) in the context of CATE estimation and can be downloaded from Prof. Leili's website (`http://www.personal.ceu.hu/staff/Robert_Lieli/cate-birthdata.zip`). We focus on White and first-time mothers, and form a random sub-sample with sample size $n = 5000$. The outcome Y is birth weight measured in grams and the treatment T is a binary indicator of maternal smoking (1: Yes, 0: No). We are interested in estimating the distribution of birth weight had all versus none of the mothers smoked in the entire population, as well as the treatment effect of maternal smoking on different birth weight quantiles. To ensure **SITA** holds, we choose a large set of variables as $\boldsymbol{X}$, including mother's age, education level (in years), the number of prenatal visits, the number of prenatal visits within the first trimester; and indicators for baby's gender, mother's marital status, gestational diabetes, hypertension, amniocentesis, ultrasound exams and alcohol use.

The response variable and all continuous covariates are mapped to the unit interval using min-max normalization. We use five posterior samples from BART probit as an informative prior for the PS, and fit the proposed model based on NN with one or two hidden layers to the data. We run the MCMC for 10000 iterations and save every 10th iteration after discarding 1000 iterations as burn-in. The number of basis functions and hidden neurons are selected from $K = \{8, 10, 12\}$, $V_1 = \{5, 8, 10, 20\}$, $V_2 = \{5, 8, 10, 20\}$ using WAIC. Fig. 4.7 shows the estimated counterfactual birth weight densities using $K = 8$ and $V_1 = 10$, along with 95% pointwise CIs, overlaid on the histogram of the observed data. The plot shows that the estimated densities fit the data well. Compared to the birth weight density of nonsmoking mothers, the birth weight density of smoking mothers is more left-skewed with significant higher density in the lower-tail range (< 3000 grams), indicating that smoking mothers have a higher probability of giving birth to infants with low birth

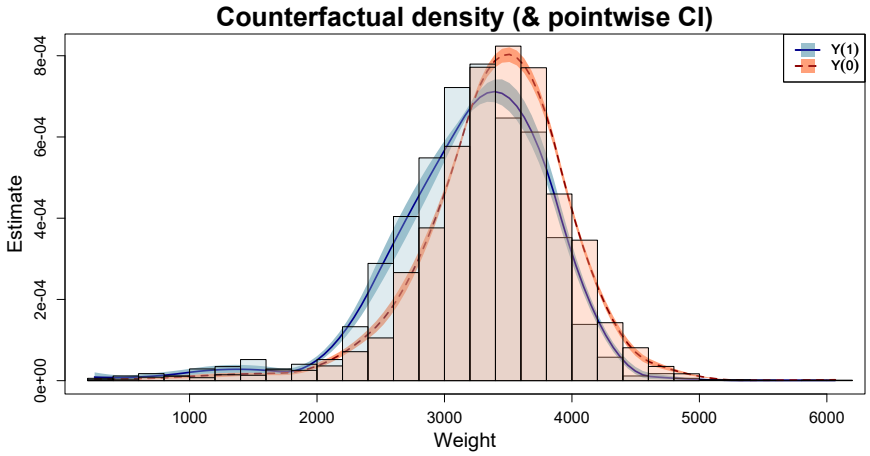


Figure 4.7: Estimated counterfactual birth weight density for infants born to smoking ($T = 1$) and nonsmoking ($T = 0$) mothers.

weight.

To more precisely understand how maternal smoking affect the birth weight distribution, we estimate its QTE at quantile levels $\tau = 0.15, 0.20, ..., 0.85$. The QTEs estimated using posterior samples associated with each of the five estimated PS, as well as the QTE estimated using all posterior samples, are plotted in Fig. 4.8. The plots show that the estimated QTEs are consistent across different estimates of the PS, indicating convergence of the proposed approach. The estimated QTE seems to be monotonically increasing across the quantile domain, indicating that maternal smoking has the most significant effect on the lower tail of the distribution, which is the most critical aspect of the distribution. The 15th QTE of maternal smoking is -236 gram (95% CI: $-188.2, -297.8$), suggesting that maternal smoking leads to a 236-gram decrease in the 15th quantile of the birth weight distribution.

4.8 Discussion

In this chapter, we proposed a Bayesian semiparametric model that allows simultaneous estimation of any functionals of the counterfactual distributions in presence of confounding. Differ from existing works that consider regression adjustment for either the scalar PS or the full-vector of covariates, we proposed to adjust for a double-balancing score that augments PS adjustment with individual covariates to fully utilize the observed information. We

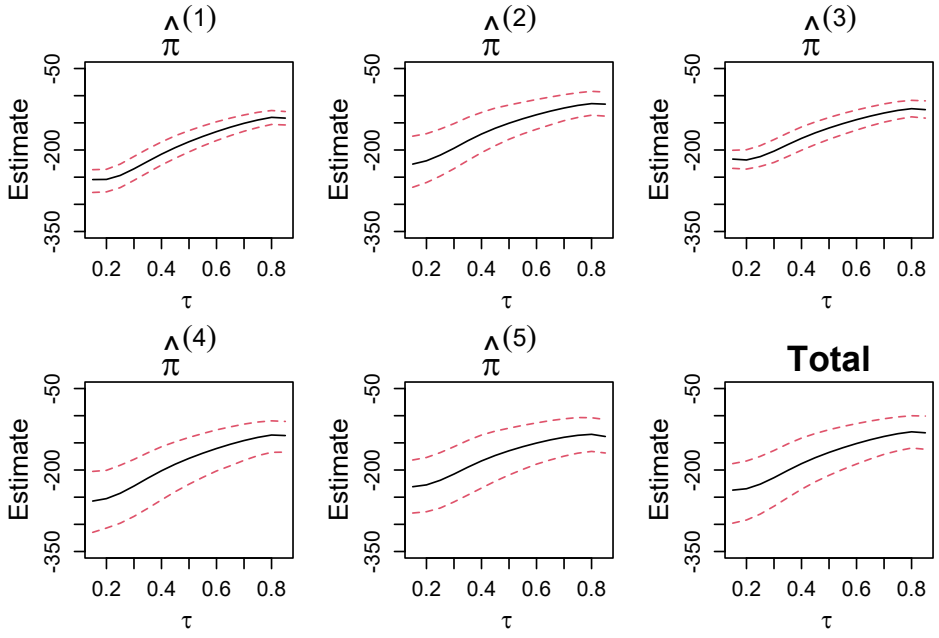


Figure 4.8: Treatment effect of maternal smoking on birth weight quantiles.

then outlined a Bayesian inference framework that incorporates uncertainty of estimating the PS, the outcome distribution conditional on the balancing score, and the covariate distribution. Carefully designed simulations provides empirical evidence that the use of double-balancing score for confounding adjustment improves performance over adjusting for any individual score alone. First, the additional PS adjustment can reduce any residual confounding if the outcome-covariate relationship is complex and not fitted well. Second, the covariate adjustment can further control the variability of the outcome residual after adjusting for the PS, rendering the estimator more efficient. In regards to counterfactual distribution estimation, the proposed approach correctly estimates PDFs and CDFs of various shapes and achieves comparable and sometimes even better performance than the state-of-art DPM of normals. Furthermore, the use of SPQR as the conditional distribution estimator allows a broader model coverage than DPM of normals and better boundary properties than truncated series estimator. In regards to QTE estimation, the proposed model consistently yields the lowest bias among all competing models.

We applied the proposed approach to estimate the causal effect of maternal smoking on infants' birth weight. The estimated counterfactual densities show that smoking leads to substantial left-skewness of the counterfactual distribution and higher probability of

giving birth to infants with low birth weight. The estimated QTEs suggest that the effect of smoking is most significant on the lower-tail quantiles of the birth weight distribution, further increasing the probability of having underweight infants. The results are consistent with the findings from other studies that maternal smoking might leads to lower birth weight (Abrevaya et al. 2015; Huang and Yang 2020; Xie et al. 2020).

Since the proposed approach always includes the full-vector of covariates, it ultimately will suffer from curse of dimensionality. A possible improvement could be to apply sufficient dimension reduction (SDR; Ma and Zhu 2013) to the covariates and construct the double-balancing score by augmenting the PS with the projected covariates. However, incorporating the uncertainty of SDR can be computationally challenging. Another possible extension is to come up with a joint estimation framework for the PS and the outcome distribution. As mentioned in Section 4.5.3, we chose a sequential approach purely for the sake of convenience. This is fundamentally different from the reason behind the work of Xu et al. (2018) where the PS has to be estimated separately to reduce bias arising from feedback issue (Zigler et al. 2013). We are also working on extending our work to estimate partial CQTE (Zhou et al. 2021).

CHAPTER

5

SINGLE-INDEX QUANTILE REGRESSION WITH MISSING AT RANDOM DATA

5.1 Background

Single-index models (Ichimura 1993; Hardle et al. 1993) have gained increasing popularity in conditional quantile estimation. They assume the τ-th conditional quantile function of a scalar response Y given a p-dimensional covariates $\boldsymbol{Z}$ has a single-index structure

$$G_\tau(\boldsymbol{z}) := Q_Y(\tau|\boldsymbol{Z}=\boldsymbol{z}) = g_{0,\tau}(\boldsymbol{z}^\top \boldsymbol{\beta}_{0,\tau}), \tag{5.1}$$

where $g_{0,\tau}(\cdot)$ is an unknown univariate link function and $\boldsymbol{\beta}_{0,\tau}$ is the unknown index vector. As a semiparametric model, the single-index QR defined in (5.1) has many advantages over parametric and nonparametric QR: (i) the single-index structure reduces the dimensionality of multivariate covariates to a univariate index $\boldsymbol{z}^\top \boldsymbol{\beta}_{0,\tau}$, effectively overcoming “curse of the dimensionality" while capturing important features in high-dimensional data; (ii) the unspecified link function allows model flexibility when $G_\tau(\boldsymbol{z})$ is strongly nonlinear,

reducing the risk of misspecification; (iii) interpreting covariate effects is as easy as plotting $g_{0,\tau}(\boldsymbol{z}^\top\boldsymbol{\beta}_{0,\tau})$ against $\boldsymbol{z}^\top\boldsymbol{\beta}_{0,\tau}$ or examining the values of the index parameters. Consequently, single-index QR has received extensive attention in the literature in the recent years. To estimate the parameters in single-index QR, namely the parametric index $\boldsymbol{\beta}_{0,\tau}$ and the nonparametric link function $g_{0,\tau}(\cdot)$, Wu et al. (2010) proposed an iterative algorithm based on local linear regression, which is similar to the minimum average variance estimation (MAVE; Xia et al. 2002) algorithm in the mean regression context. Later, Kong and Xia (2012) proposed a refined algorithm by introducing a penalty term that improves convergence. Christou and Akritas (2016) proposed a non-iterative algorithm that relies on a first-stage nonparametric estimator. Ma and He (2016) proposed a pseudo-profile likelihood approach using spline approximation. Hu et al. (2013) adopted a fully Bayesian framework using Gaussian process and asymmetric Laplace working likelihood. Recently, Jiang and Yu (2021) considered single-index QR under non-crossing constraints. These papers all assume that the observations are complete.

Missing data are a frequently encountered problem in many applications, and complete-case (CC) analysis that simply discards all incomplete observations can result in biased and inefficient estimators. When data are classified as missing at random (MAR; Little and Rubin 2019), meaning that the missingness mechanism depends only on the observed data and not on the unobserved data, three established missing data handling mechanisms are commonly seen in the literature. (i) The IPW method tries to correct for selection bias by reweighting the CC observations with the inverse of the propensity score (PS, the probability of being a respondent given the observed data). Standard statistical inference can then be performed on the weighted data. In the context of QR, Sherwood et al. (2013) studied the inverse PS weighted QR estimator when covariates are missing. Chen et al. (2015) adopted the same approach and extended it to the general MAR setting under which the response may also be missing. Zhou et al. (2021) combined IPW and local linear QR for nonparametric estimation of conditional quantile treatment effect (which can be considered as a special case of QR with MAR outcomes). Although the IPW QR estimator is easy to calculate thanks to its natural connection with weighted QR, its consistency depends on a consistent estimator for the PS and thus may be biased when the PS model is misspecified. Furthermore, the IPW method does not fully use the information of the observed variables in the incomplete observations, and thus the resulting estimator might be inefficient. (ii) The estimating equations projection (EEP) method (Zhou et al. 2008) augments the CC estimating equations with a projection term that draws information from

the distribution of missing variables conditional on the observed variables. Its statistical properties in the context of QR have been studied by Wei and Yang (2014) and Chen et al. (2015). Compared to the IPW method, the EEP method is less prone to misspecification since the augmentation term is often modeled nonparametrically using kernel regression. However, it does not use any information from the missing mechanism. (iii) The augmented IPW (AIPW) method adjusts the EEP method by estimated value of the inverse PS. It can be seen as an amalgamation of IPW and EEP methods through the estimating equations. By fully exploiting the information from the observed variables, the AIPW method aims to retain the best of and improve on both IPW and EEP approaches. Chen et al. (2015) established the results for AIPW adjusted estimating equations for linear QR. Wang et al. (2022) later extended it to nonparametric QR model using local estimating equations.

Despite abundant literature on QR with missing data, only scant attention has been paid to single-index QR when the data contain missing values. This is partly due to the inherent complexity that results from combining single-index model and QR. Compared to parametric or nonparametric regression, single-index regression requires solving two optimization problems or estimating equations and thus is more computationally challenging. The problem is further complicated by the nonsmooth nature of QR objective functions/estimating equations, which renders traditional differential-based optimization methods inapplicable. Zou et al. (2020) defined weighted estimators of index parameters and link function for partially linear single-index QR when the response is censored and the censoring indicator is MAR. Liang et al. (2021) focused on the general MAR data setting, and Liu and Liang (2022) developed a Bayesian framework. It is worth noting that all existing methods focused on the IPW method. However, as mentioned previously, the IPW method is prone to model misspecification, and the addition of a projection term often improves efficiency of the resulting estimator.

5.2 Contribution

In this chapter, we develop a general framework for estimating parameters in single-index QR when the data is MAR, which allows the response or the covariates or both to be missing. In particular, by using spline approximation of the nonparametric link function and profile estimation principle, we develop a class of weighted pseudo-likelihood estimators that include the IPW estimator, the EEP estimator, and the AIPW estimator, which can computed

using an efficient algorithm. In addition, we show that the EEP and AIPW approaches are asymptotically equivalent to a nonparametric IPW approach, bridging and providing an intuitive interpretation of the nuanced differences between the three approaches. To our best knowledge, this is the first work in the literature that considers EEP and AIPW approaches for estimating single-index model parameters. Simulation studies show that the proposed methods can effectively reduce estimation bias of the nonparametric link function when covariates are MAR, and improve numerical stability of the estimation algorithm in general. In addition, we show that the EEP approach is generally more robust and efficient than the IPW approach; and the AIPW approach is robust to either misspecification of the PS model or poor estimation of the conditional distribution of missing variables. Finally, we illustrate our methodology by estimating the single-index relationship between percentage body fat and body circumference measurements when the covariates are MAR.

5.3 Organization

The rest of the chapter is organized as follows. In Section 5.4, we introduce the weighted pseudo-likelihoods for estimating single-index QR parameters when observations are MAR. In Section 5.5, we outline an efficient algorithm for optimizing the pseudo-likelihoods and provide some insights on parameter selection. Simulation studies demonstrating improved performance over the CC estimator is presented in Section 5.6. We apply the proposed estimation framework to the body fat data set in Section 5.7 and conclude the chapter with some discussion in Section 5.8.

5.4 Single-index QR and missing data

Let $\{Y_i, \boldsymbol{Z}_i\}_{i=1}^n$ with $\boldsymbol{Z}_i \in \mathscr{D} \subset \mathbb{R}^p$ be an observed sample from $(Y, \boldsymbol{Z})$. For any given quantile level τ, we assume the conditional quantile has a single-index structure as defined in (5.1), which can be conveniently expressed as

$$Y_i = g_{0,\tau}(\boldsymbol{Z}_i^\top \boldsymbol{\beta}_{0,\tau}) + \epsilon_i$$

where ϵ_i is the random error term satisfying $P(\epsilon_i \leq 0 | \boldsymbol{Z}_i) = \tau$. We assume that the covariates $\boldsymbol{Z}_i$'s are independent and identically distributed (i.i.d.), and ϵ_i's are independent which

nests the i.i.d. situation as a special case. Hereafter, for the convenience of notation, we omit the subscript τ and use $g_0(\cdot) \equiv g_{0,\tau}(\cdot)$, $\boldsymbol{\beta}_0 \equiv \boldsymbol{\beta}_{0,\tau}$ whenever there is no confusion. Due to the nonparametric nature of $g_0(\cdot)$, the scale of $\boldsymbol{\beta}_0$ is not identifiable. Throughout the chapter, we assume that $\boldsymbol{\beta}_0$ is an inner point of the parameter space

$$\mathscr{B} = \{\boldsymbol{\beta} = (\beta_1, ..., \beta_p)^\top : \|\boldsymbol{\beta}\|_2 = 1, \beta_r > 0, \boldsymbol{\beta} \in \mathbb{R}^p\}$$

where $\|\cdot\|_2$ denotes the Euclidean norm and β_r is the first nonzero element of $\boldsymbol{\beta}$. Such choice of $\mathscr{B}$ is common in single-index model literature (Yu and Ruppert 2002; Zhu and Xue 2006; Wu et al. 2010).

The goal of this chapter is to estimate both $\boldsymbol{\beta}_0$ and $g_0(\cdot)$ when some observations may be missing. Let $(\boldsymbol{X}_i^\top, \boldsymbol{X}_i^{c\top})^\top$ be the vector formed by rearranging the elements of $(Y_i, \boldsymbol{Z}_i^\top)^\top$ such that $\boldsymbol{X}_i$, a d-dimensional nonnull vector with $0 < d \leq p$, is observed for all i's, while $\boldsymbol{X}_i^{\mathsf{c}}$ contains elements for which observations may be missing for some i's. Under this unifying notation, the missing observations may be induced from the response, the covariates, or both. In this chapter, we focus on two of such cases: (i) $\boldsymbol{X}_i = Y_i$ such that data are missing from all of the covariates but fully observed for the response; and (ii) $\boldsymbol{X}_i = \boldsymbol{Z}_i$ such that data are missing from the response only. Some discussions on how to handle other missing data scenarios using the proposed method are given in Section 5.8.

Let δ_i be an indicator variable such that $\delta_i = 1$ if all values in $\boldsymbol{X}_i^{\mathsf{c}}$ are observed and $\delta_i = 0$ otherwise, the observed data can then be denoted as $(\boldsymbol{X}_i, \delta_i, \boldsymbol{X}_i^{\mathsf{c}} : 1 \leq i \leq n)$. We assume that $\boldsymbol{X}_i^{\mathsf{c}}$ are MAR (Little and Rubin 2019). Under the MAR assumption, the PS (propensity score) depends only on the observed data

$$\pi_{i0} \equiv \pi_0(\boldsymbol{X}_i) = P(\delta_i = 1 | Y_i, \boldsymbol{Z}_i) = P(\delta_i = 1 | \boldsymbol{X}_i).$$

In practice, the true PS function is often unknown and needs to be estimated. To avoid "curse of dimensionality" we assume a parametric model for the PS, i.e.,

$$\pi_{i0} = \pi(\boldsymbol{X}_i; \boldsymbol{\gamma}),$$

where $\pi(\cdot)$ is a known smooth function and $\boldsymbol{\gamma} \in \mathbb{R}^d$ is an unknown finite-dimensional parameter. For example, the logistic regression model $\pi_{i0} = \text{expit}(\gamma_1 + \boldsymbol{X}_i^\top \boldsymbol{\gamma}_2)$ with $\boldsymbol{\gamma} = (\gamma_1, \boldsymbol{\gamma}_2)^\top$ is a natural choice for modeling the conditional expectation of a binary variable. Let $\hat{\boldsymbol{\gamma}}$ be the maximum likelihood estimator (MLE) of $\boldsymbol{\gamma}_0$, then $\hat{\pi}_i \equiv \pi(\boldsymbol{X}_i; \hat{\boldsymbol{\gamma}})$ is a parametric

estimator of π_{i0}.

5.4.1 Profile pseudo-likelihood estimation

To motivate the proposed framework, we consider the case when $\delta_i = 1$, $i = 1, ..., n$, such that the data are fully observed. Let $U(\boldsymbol{z}; \boldsymbol{\beta}) = \boldsymbol{z}^\top \boldsymbol{\beta}$ be the linear index with support $\mathscr{U} = \{\boldsymbol{z}^\top \boldsymbol{\beta}, \boldsymbol{z} \in \mathscr{C}, \boldsymbol{\beta} \in \mathscr{B}\}$. We define the support of $g_0(\cdot)$ as $[a, b]$ where $a = \inf(\mathscr{U})$ and $b = \sup(\mathscr{U})$. For any given $\boldsymbol{\beta} \in \mathscr{B}$ and $u \in \mathscr{U}$, we define $g(u; \boldsymbol{\beta})$ to be the τth quantile function of Y given $\boldsymbol{Z}^\top \boldsymbol{\beta} = u$ such that $g(\boldsymbol{Z}^\top \boldsymbol{\beta}_0; \boldsymbol{\beta}_0) = g_0(\boldsymbol{Z}^\top \boldsymbol{\beta}_0)$. We use the notation $g(\cdot; \boldsymbol{\beta})$ to emphasize the implicit dependence of $g(\cdot)$ on $\boldsymbol{\beta}$ through estimation, which will become apparent later. By definition, the true parameter vector $\boldsymbol{\beta}_0$ is the unique solution to the minimization problem

$$\boldsymbol{\beta}_0 = \underset{\boldsymbol{\beta} \in \mathscr{B}}{\arg\min}\, \mathbb{E}\left[\rho_\tau \left\{ Y - g(\boldsymbol{Z}^\top \boldsymbol{\beta}; \boldsymbol{\beta}) \right\}\right].$$

We use polynomial splines to approximate the nonparametric function $g(\cdot)$. Let $a = u_0 < u_1 < \cdots < u_{k_n+1} = b$ be a partition of $[a, b]$ with k_n equally spaced internal knots. Let $\boldsymbol{B} = \{B_j(u) : 1 \le j \le J_n\}$ be a normalized B-spline basis functions on $[a, b]$ of order m with k_n internal knots such that $J_n = k_n + m$. The quantile function can be approximated well by a B-spline function such that $g_0(\boldsymbol{z}^\top \boldsymbol{\beta}_0) \approx \boldsymbol{B}(\boldsymbol{z}^\top \boldsymbol{\beta}_0)^\top \boldsymbol{\theta}_0$ for some $\boldsymbol{\theta}_0 \in \mathbb{R}^{J_n}$. The estimators of the spline coefficients $\boldsymbol{\theta}_0$ and the index parameters $\boldsymbol{\beta}_0$ are taken to be the minimizers of the following pseudo-likelihood function

$$L_{\tau n}(\boldsymbol{\theta}, \boldsymbol{\beta}) = n^{-1} \sum_{i=1}^{n} \rho_\tau \left\{ Y_i - \boldsymbol{B}(\boldsymbol{Z}_i^\top \boldsymbol{\beta})^\top \boldsymbol{\theta} \right\}.$$

Following Ma and He (2016), we define the profile pseudo-likelihood function of $\boldsymbol{\beta}$ as

$$\tilde{L}_{\tau n}(\boldsymbol{\beta}) = L_{\tau n}(\tilde{\boldsymbol{\theta}}(\boldsymbol{\beta}), \boldsymbol{\beta}) = n^{-1} \sum_{i=1}^{n} \rho_\tau \left\{ Y_i - \boldsymbol{B}(\boldsymbol{Z}_i^\top \boldsymbol{\beta})^\top \tilde{\boldsymbol{\theta}}(\boldsymbol{\beta}) \right\}.$$

where $\tilde{\boldsymbol{\theta}}(\boldsymbol{\beta})$ is the minimizer of $L_{\tau n}(\boldsymbol{\theta}, \boldsymbol{\beta})$ over $\boldsymbol{\theta} \in \mathbb{R}^{J_n}$ for given $\boldsymbol{\beta}$. The profile estimator of $\boldsymbol{\beta}_0$ is then taken as

$$\hat{\boldsymbol{\beta}} = \underset{\boldsymbol{\beta} \in \mathscr{B}}{\arg\min} \left\{ \tilde{L}_{\tau n}(\boldsymbol{\beta}) \right\},$$

and the spline estimator $\hat{g}(u)$ of $g_0(u)$ for any $u \in \mathscr{U}$ is simply $g_n(u; \hat{\boldsymbol{\beta}}) = \boldsymbol{B}(u)^\top \hat{\boldsymbol{\theta}}(\hat{\boldsymbol{\beta}})$ where $\hat{\boldsymbol{\theta}}(\hat{\boldsymbol{\beta}})$ minimizes $L_{\tau n}(\boldsymbol{\theta}, \hat{\boldsymbol{\beta}})$.

When the data are fully observed, $\hat{\boldsymbol{\beta}}$ and $\hat{g}(u)$ are consistent for $\boldsymbol{\beta}_0$ and $g_0(u)$, respectively (Ma and He 2016). However, the same approach cannot be employed when the data are MAR since the objective functions involve missing values. A naive workaround is to estimate $\boldsymbol{\beta}_0$ and $g_0(u)$ using the CC data $\{(Y_i, \boldsymbol{Z}_i) : \delta_i = 1\}$ only. When only the response is MAR but all covariates are fully observed, the CC estimator remains consistent since the distribution of $Y|\boldsymbol{X}$ is the same as that of $Y|\boldsymbol{X}, \delta = 1$. When the covariates are MAR and the missing probability mechanism depends on the response, however, the CC estimator can be seriously biased.

We next describe our three proposed methods for handling MAR observations in the single-index QR context.

5.4.2 IPW method

Under MAR assumption, we have

$$\mathbb{E}\left[\frac{\delta}{\pi_0(\boldsymbol{X})}\rho_\tau\left\{Y - g(\boldsymbol{Z}^\top\boldsymbol{\beta};\boldsymbol{\beta})\right\}\right] = \mathbb{E}\left[\rho_\tau\left\{Y - g(\boldsymbol{Z}^\top\boldsymbol{\beta};\boldsymbol{\beta})\right\}\right].$$

Thus, by weighting $\tilde{L}_{\tau n}(\boldsymbol{\beta})$ by the inverse of estimated PS, the IPW estimator for $\boldsymbol{\beta}_0$ is defined as the minimizer of the IPW adjusted pseudo-likelihood function

$$\tilde{L}^{\mathrm{IPW}}_{\tau n}(\boldsymbol{\beta}) = \sum_{i=1}^{n} \delta_i W_i^{\mathrm{IPW}} \rho_\tau\left\{Y_i - \boldsymbol{B}(\boldsymbol{Z}_i^\top\boldsymbol{\beta})^\top\tilde{\boldsymbol{\theta}}^{\mathrm{IPW}}(\boldsymbol{\beta})\right\}, \tag{5.2}$$

where $W_i^{\mathrm{IPW}} = 1/\hat{\pi}_i$ and $\tilde{\boldsymbol{\theta}}^{\mathrm{IPW}}(\boldsymbol{\beta})$ minimizes

$$L^{\mathrm{IPW}}_{\tau n}(\boldsymbol{\theta},\boldsymbol{\beta}) = n^{-1}\sum_{i=1}^{n} \delta_i W_i^{\mathrm{IPW}} \rho_\tau\left\{Y_i - \boldsymbol{B}(\boldsymbol{Z}_i^\top\boldsymbol{\beta})^\top\boldsymbol{\theta}\right\}$$

over $\boldsymbol{\theta} \in \mathbb{R}^{J_n}$. Analogously, the IPW estimator $\hat{g}^{\mathrm{IPW}}(u)$ of $g_0(u)$ is taken as $g_n^{\mathrm{IPW}}(u;\hat{\boldsymbol{\beta}}^{\mathrm{IPW}}) = \boldsymbol{B}^\top\hat{\boldsymbol{\theta}}^{\mathrm{IPW}}(\hat{\boldsymbol{\beta}}^{\mathrm{IPW}})$ where $\hat{\boldsymbol{\theta}}^{\mathrm{IPW}}(\hat{\boldsymbol{\beta}}^{\mathrm{IPW}})$ minimizes $L_{\tau n}(\boldsymbol{\theta},\hat{\boldsymbol{\beta}}^{\mathrm{IPW}})$.

The IPW estimators based on (5.2) are consistent whenever the PS are estimated consistently (Liu and Liang 2022). In practice, however, finding a correct parametric model $\pi(\cdot;\gamma)$ for the PS is a challenging task, and consequently the IPW estimators are vulnerable to misspecification. Furthermore, information of the fully observed variables is only used in estimating the PS but not directly in estimating the single-index parameters, and thus

the IPW estimators might not be efficient enough. This calls for an approach that directly incorporates the observed variables in the objective functions.

5.4.3 EEP method

Let $S(Y,\boldsymbol{Z},\boldsymbol{\beta}) = \phi_\tau\left\{Y - g(\boldsymbol{Z}^\top\boldsymbol{\beta};\boldsymbol{\beta})\right\}\partial g(\boldsymbol{Z}^\top\boldsymbol{\beta};\boldsymbol{\beta})/\partial\boldsymbol{\beta}$ be the original estimating function for single-index regression quantiles, where $\phi_\tau(u) = \tau - 1(u<0)$ is the subgradient of the check function. If the data are fully observed for all subjects, the QR estimator of $\boldsymbol{\beta}_0$ can be obtained by solving the estimating equation

$$\sum_{i=1}^{n} S(Y_i,\boldsymbol{Z}_i,\boldsymbol{\beta}) \approx 0.$$

We use '$\approx$' as in Chen et al. (2015) to indicate that the exact solution may not exist because of the nonsmoothness of the function ϕ. Consider the following modified estimating function to handle MAR observations

$$S_i^{\mathrm{EEP}}(\hat{m},\boldsymbol{\beta}) = \delta_i S(Y_i,\boldsymbol{Z}_i,\boldsymbol{\beta}) + (1-\delta_i)\hat{m}(\boldsymbol{X}_i,\boldsymbol{\beta}), \tag{5.3}$$

where $\hat{m}(\boldsymbol{X}_i,\boldsymbol{\beta})$ is an estimator of $m(\boldsymbol{X}_i,\boldsymbol{\beta}) = \mathbb{E}[S(Y_i,\boldsymbol{Z}_i,\boldsymbol{\beta})|\boldsymbol{X}_i]$. Clearly, if $\hat{m}(\boldsymbol{X}_i,\boldsymbol{\beta})$ is consistent, solving the estimating equation

$$\sum_{i=1}^{n} S_i^{\mathrm{EEP}}(\hat{m},\boldsymbol{\beta}) \approx 0. \tag{5.4}$$

leads to a valid estimator for $\boldsymbol{\beta}_0$ since $\mathbb{E}\left[n^{-1}\sum_{i=1}^{n} S_i^{\mathrm{EEP}}(\hat{m},\boldsymbol{\beta})\right] \approx 0$. The solution to (5.4) is often referred to as the EEP estimator of $\boldsymbol{\beta}$ since the conditional expectation $\mathbb{E}[S(Y_i,\boldsymbol{Z}_i,\boldsymbol{\beta})|\boldsymbol{X}_i]$ can be interpreted as the projection of $S(Y_i,\boldsymbol{Z}_i,\boldsymbol{\beta})$ into the space generated by the fully observed data (Zhou et al. 2008). Under the MAR assumption, the missing indicator δ is conditionally independent of $\boldsymbol{X}^{\mathrm{c}}$ given $\boldsymbol{X}$. It then follows that

$$\mathbb{E}\{S(Y,\boldsymbol{Z},\boldsymbol{\beta})|\boldsymbol{X}\} = \mathbb{E}\{S(Y,\boldsymbol{Z},\boldsymbol{\beta})|\delta=1,\boldsymbol{X}\},$$

which implies that $m(\boldsymbol{X}_i,\boldsymbol{\beta})$ can be consistently estimated by using the CC observations $\{(Y_i,\boldsymbol{Z}_i) : \delta_i = 1\}$ only. Following Zhou et al. (2008), we estimate $m(\boldsymbol{X}_i,\boldsymbol{\beta})$ using a kernel

smoother

$$\hat{m}(\boldsymbol{X}_i,\boldsymbol{\beta})=\frac{\sum_{j=1}^{n}\mathscr{K}_h(\boldsymbol{X}_j-\boldsymbol{X}_i)\delta_j S(Y_j,\boldsymbol{X}_j,\boldsymbol{\beta})}{\sum_{j=1}^{n}\mathscr{K}_h(\boldsymbol{X}_j-\boldsymbol{X}_i)\delta_j} \tag{5.5}$$

where $\mathscr{K}_h(\cdot)=\mathscr{K}(\cdot/h)/h^d$, $\mathscr{K}(\cdot)$ is a d-dimensional kernel function and $h\equiv h_n\to 0$ is a bandwidth parameter. Note that in order for the kernel estimator defined above to possess justifiable asymptotic properties, $\boldsymbol{X}_i$ are in fact required to be i.i.d. samples from the population $\boldsymbol{X}$ (Chen et al. 2015). Therefore the EEP approach cannot handle the scenario when $\boldsymbol{Z}_i$'s are missing and ϵ_i's are not i.i.d.

Substituting $\hat{m}(\boldsymbol{X}_i,\boldsymbol{\beta})$ into (5.4), Equation (5.4) can be expanded as

$$\sum_{i=1}^{n}\delta_i S(Y_i,\boldsymbol{Z}_i,\boldsymbol{\beta})+\sum_{i=1}^{n}\sum_{j=1}^{n}(1-\delta_i)\frac{\mathscr{K}_h(\boldsymbol{X}_j-\boldsymbol{X}_i)\delta_j S(Y_j,\boldsymbol{X}_j,\boldsymbol{\beta})}{\sum_{j=1}^{n}\mathscr{K}_h(\boldsymbol{X}_j-\boldsymbol{X}_i)\delta_j}\approx 0. \tag{5.6}$$

By swapping the summation index of the second component on the left hand side, the expression above can be rewritten as

$$\sum_{i=1}^{n}\delta_i W_i^{\mathrm{EEP}} S(Y_i,\boldsymbol{Z}_i,\boldsymbol{\beta})\approx 0$$

where

$$W_i^{\mathrm{EEP}}=1+\sum_{j=1}^{n}(1-\delta_j)\frac{\mathscr{K}_h(\boldsymbol{X}_j-\boldsymbol{X}_i)}{\sum_{i=1}^{n}\mathscr{K}_h(\boldsymbol{X}_i-\boldsymbol{X}_j)\delta_i}.$$

Therefore, the modified estimating equation for single-index regression quantiles under EEP approach is simply a weighted variant of the original estimating equation. Using the spline approximation $g(\boldsymbol{z}^\top\boldsymbol{\beta}_0)\approx\boldsymbol{B}(\boldsymbol{z}^\top\boldsymbol{\beta}_0)^\top\boldsymbol{\theta}_{\mathbf{0}}$ and the fact that $W_i^{\mathrm{EEP}}\geq 0$ for all i, we can construct the EEP profile pseudo-likelihood function for $\boldsymbol{\beta}$ as

$$\tilde{L}_{\tau n}^{\mathrm{EEP}}(\boldsymbol{\beta})=\sum_{i=1}^{n}\delta_i W_i^{\mathrm{EEP}}\rho_\tau\left\{Y_i-\boldsymbol{B}(\boldsymbol{Z}_i^\top\boldsymbol{\beta})^\top\tilde{\boldsymbol{\theta}}^{\mathrm{EEP}}(\boldsymbol{\beta})\right\},$$

where $\tilde{\boldsymbol{\theta}}^{\mathrm{EEP}}$ minimizes

$$L_{\tau n}^{\mathrm{EEP}}(\boldsymbol{\theta},\boldsymbol{\beta})=n^{-1}\sum_{i=1}^{n}\delta_i W_i^{\mathrm{EEP}}\rho_\tau\left\{Y_i-\boldsymbol{B}(\boldsymbol{Z}_i^\top\boldsymbol{\beta})^\top\boldsymbol{\theta}\right\}$$

over $\boldsymbol{\theta}\in\mathbb{R}^{J_n}$. Analogously, the EEP estimator $\hat{g}^{\mathrm{EEP}}(u)$ of $g_0(u)$ is then taken as $g_n^{\mathrm{EEP}}(u;\hat{\boldsymbol{\beta}}^{\mathrm{EEP}})=$

$\boldsymbol{B}^\top \hat{\boldsymbol{\theta}}^{\text{EEP}}(\hat{\boldsymbol{\beta}}^{\text{EEP}})$ where $\hat{\boldsymbol{\theta}}^{\text{EEP}}(\hat{\boldsymbol{\beta}}^{\text{EEP}})$ minimizes $L^{\text{EEP}}_{\tau n}(\boldsymbol{\theta}, \hat{\boldsymbol{\beta}}^{\text{EEP}})$.

Compared to the IPW approach, the EEP approach directly estimates the distribution of missing outcomes conditional on the observed covariates using nonparametric regression. Therefore, it is less vulnerable to model misspecfication and may be more efficient.

5.4.4 AIPW method

The AIPW method is an amalgamation of the IPW method and the EEP method via the estimating function

$$S_i^{\text{AIPW}}(\hat{m}, \hat{\pi}, \boldsymbol{\beta}) = \frac{\delta_i}{\hat{\pi}_i} S(Y_i, \boldsymbol{Z}_i, \boldsymbol{\beta}) + \left(1 - \frac{\delta_i}{\hat{\pi}_i}\right) \hat{m}(\boldsymbol{X}_i, \boldsymbol{\beta}). \tag{5.7}$$

It weighs the original estimating function and its projection by IPW and 1 minus IPW, respectively. Note that (5.7) can be written equivalently as

$$\hat{m}(\boldsymbol{X}_i, \boldsymbol{\beta}) + \frac{\delta_i}{\hat{\pi}_i} \{S(Y_i, \boldsymbol{Z}_i, \boldsymbol{\beta}) - \hat{m}(\boldsymbol{X}_i, \boldsymbol{\beta})\},$$

which can be viewed as a robustified version of $\hat{m}(\boldsymbol{X}_i, \boldsymbol{\beta})$ with an IPW bias correction term. Therefore, even if the kernel smoother estimates the projection space poorly, the bias can be estimated accurately and help improve overall estimation as long as the PS are estimated consistently. Let $\hat{\boldsymbol{\beta}}^{\text{AIPW}}$ be the AIPW estimator of $\boldsymbol{\beta}_0$, then $\hat{\boldsymbol{\beta}}^{\text{AIPW}}$ solves

$$\sum_{i=1}^{n} S_i^{\text{AIPW}}(\hat{m}, \hat{\pi}, \boldsymbol{\beta}) \approx 0.$$

Using similar steps as in (5.6), we can rewrite the AIPW adjusted estimating equation as

$$\sum_{i=1}^{n} \delta_i W_i^{\text{AIPW}} S(Y_i, \boldsymbol{X}_i, \boldsymbol{\beta}) \approx 0,$$

where

$$W_i^{\text{AIPW}} = W_i^{\text{IPW}} + \sum_{j=1}^{n} (1 - \delta_j W_j^{\text{IPW}}) \frac{\mathscr{K}_h(\boldsymbol{X}_j - \boldsymbol{X}_i)}{\sum_{i=1}^{n} \mathscr{K}_h(\boldsymbol{X}_i - \boldsymbol{X}_j)\delta_i}$$

Therefore, the AIPW adjusted estimating equation is also a weighted variant of the original estimating equation. If we assume $W_i^{\text{AIPW}} \geq 0$ for all i for the moment, then by using spline

approximation we can construct the AIPW profile pseudo-likelihood function for $\boldsymbol{\beta}$ as

$$\tilde{L}_{\tau n}^{\mathrm{AIPW}}(\boldsymbol{\beta}) = \sum_{i=1}^{n} \delta_i W_i^{\mathrm{AIPW}} \rho_\tau \left\{ Y_i - \boldsymbol{B}(\boldsymbol{Z}_i^\top \boldsymbol{\beta})^\top \tilde{\boldsymbol{\theta}}^{\mathrm{AIPW}}(\boldsymbol{\beta}) \right\}, \tag{5.8}$$

where $\tilde{\boldsymbol{\theta}}^{\mathrm{AIPW}}$ minimizes

$$L_{\tau n}^{\mathrm{AIPW}}(\boldsymbol{\theta}, \boldsymbol{\beta}) = n^{-1} \sum_{i=1}^{n} \delta_i W_i^{\mathrm{AIPW}} \rho_\tau \left\{ Y_i - \boldsymbol{B}(\boldsymbol{Z}_i^\top \boldsymbol{\beta})^\top \boldsymbol{\theta} \right\} \tag{5.9}$$

over $\boldsymbol{\theta} \in \mathbb{R}^{J_n}$. Analogously, the AIPW estimator $\hat{g}^{\mathrm{AIPW}}(u)$ of $g_0(u)$ is taken as $g_n^{\mathrm{AIPW}}(u; \hat{\boldsymbol{\beta}}^{\mathrm{AIPW}}) = \boldsymbol{B}^\top \hat{\boldsymbol{\theta}}^{\mathrm{AIPW}}(\hat{\boldsymbol{\beta}}^{\mathrm{AIPW}})$ where $\hat{\boldsymbol{\theta}}^{\mathrm{AIPW}}(\hat{\boldsymbol{\beta}}^{\mathrm{AIPW}})$ minimizes $L_{\tau n}^{\mathrm{AIPW}}(\boldsymbol{\theta}, \hat{\boldsymbol{\beta}}^{\mathrm{AIPW}})$.

The AIPW approach aims to retain the best of its parent approaches. Compared to the IPW approach, it incorporates additional information from the conditional distribution of missing variables and thus may be more efficient. It is also more robust than the IPW approach since the conditional distribution is estimated using nonparametric kernel regression. Compared to the EEP approach, the AIPW approach can take advantage of the IPW bias-correction term which provides a safeguard against pitfall of the kernel smoother.

5.4.5 A unifying framework

Our results showed that it is possible to use a unifying expression to summarize the profile pseudo-likelihood for single-index regression quantiles under IPW, EEP, and AIPW approaches, and the three approaches only differ in the weights they assign to the check loss. Specifically, the IPW method weighs each observation only by the PS; the EEP method weighs each observation only by kernel weights; and the AIPW method weighs each observation by a hybrid weight composed of both PS and kernel weights.

The underlying correlation between the three approaches is, unsurprisingly, more than just structural resemblance. In fact, as we show below, the EEP and AIPW approaches are essentially a nonparametric IPW approach, at least in an asymptotic sense, since they are also estimating the true inverse PS but with kernel regression instead of parametric model. This observation supplements the work of Chen et al. (2015) who showed that nonparametric IPW, EEP and AIPW estimating equations for linear QR are asymptotically equivalent, and offers an intuitive interpretation of the similarities and nuanced differences between the three widely used approaches for handling missing data.

Let $f(\boldsymbol{x})$ be the probability density function of $\boldsymbol{X}$, and $0 < f(\boldsymbol{x}) < \infty$ in the support of $\boldsymbol{X}$. Denote $r(\boldsymbol{x}) = f(\boldsymbol{x})\pi_0(\boldsymbol{x})$. Let $\hat{\pi}_n(\boldsymbol{x})$ denote the Nadaraya-Watson estimator of $\pi_0(\boldsymbol{x})$, defined as

$$\hat{\pi}_n(\boldsymbol{x}) = \frac{\sum_{i=1}^n \mathscr{K}_h(\boldsymbol{X}_i - \boldsymbol{x})\delta_i}{n\hat{f}_n(\boldsymbol{x})},$$

where $\hat{f}_n(\boldsymbol{x}) = \frac{1}{n}\sum_i^n \mathscr{K}_h(\boldsymbol{X}_i - \boldsymbol{x})$ is the kernel estimator of $f(\boldsymbol{x})$. Denote $\hat{r}_n(\boldsymbol{x}) = \hat{f}_n(\boldsymbol{x})\hat{\pi}_n(\boldsymbol{x})$. Similarly, let

$$\hat{m}_n(\boldsymbol{x}) = \frac{\sum_{i=1}^n \mathscr{K}_h(\boldsymbol{X}_i - \boldsymbol{x})\frac{1-\delta_i}{\hat{r}_n(\boldsymbol{X}_i)}}{n\hat{f}_n(\boldsymbol{x})}$$

be the kernel estimator of $m(\boldsymbol{x}) = \mathbb{E}\left\{\frac{1-\delta}{\hat{r}_n(\boldsymbol{X})}\middle|\boldsymbol{X} = \boldsymbol{x}\right\}$, and

$$\hat{\tilde{m}}_n(\boldsymbol{x};\pi,\gamma) = \frac{\sum_{i=1}^n \mathscr{K}_h(\boldsymbol{X}_i - \boldsymbol{x})\frac{1-\delta_i/\pi(\boldsymbol{X}_i,\gamma)}{\hat{r}_n(\boldsymbol{X}_i)}}{n\hat{f}_n(\boldsymbol{x})}$$

be the kernel estimator of $\tilde{m}(\boldsymbol{x};\pi,\gamma) = \mathbb{E}\left\{\frac{1-\delta/\pi(\boldsymbol{x};\hat{\gamma})}{\hat{r}_n(\boldsymbol{x})}\middle|\boldsymbol{X} = \boldsymbol{x}\right\}$. We also need the following notations for the theorem:

$$A_n(\boldsymbol{x}) = \frac{1}{n}\sum_{j=1}^n (1-\delta_j)\frac{\mathscr{K}_h(\boldsymbol{X}_j - \boldsymbol{x})}{\hat{r}_n(\boldsymbol{X}_j)},\ B_n(\boldsymbol{x};\pi,\hat{\gamma}) = \frac{1}{n}\sum_{j=1}^n \left(1 - \frac{\delta_j}{\pi(\boldsymbol{X}_j;\hat{\gamma})}\right)\frac{\mathscr{K}_h(\boldsymbol{X}_j - \boldsymbol{x})}{\hat{r}_n(\boldsymbol{X}_j)}.$$

We assume the following conditions.

(C1) $f(\boldsymbol{x})$ and $\pi_0(\boldsymbol{x})$ have bounded partial derivatives up to an order l with $l \geq 2$, $l > d$, $\inf_{\boldsymbol{x}} \pi_0(\boldsymbol{x}) \geq c_0$ and $\inf_{\boldsymbol{x}} r(\boldsymbol{x}) \geq \bar{c}_0$ where c_0 and $\bar{c}_0$ are positive constants.

(C2) $m(\boldsymbol{x})$ and $\tilde{m}(\boldsymbol{x};\pi,\hat{\gamma})$ have bounded partial derivatives with respect to $\boldsymbol{x}$ up to an order l.

(C3) $\mathscr{K}(\cdot)$ is a d-dimensional multivariate kernel function with compact support and order l, and the bandwidth h satisfies $h \to 0$, $n^\nu h^d \to \infty$ for some $0 < \nu < 1/2$ and $nh^{2l} \to 0$ as $n \to \infty$.

Conditions (C1)–(C3) are commonly used in missing data and kernel regression literature (Zhou et al. 2008; Chen et al. 2015; Wang et al. 2022). Together, they guarantee the uniform convergence of the kernel estimators $\hat{\pi}_n(\boldsymbol{x})$, $\hat{f}_n(\boldsymbol{x})$, $\hat{m}_n(\boldsymbol{x})$, and $\hat{\tilde{m}}_n(\boldsymbol{x};\pi,\hat{\gamma})$, respectively (Mack and Silverman 1982; Stute 1982).

Theorem 5.1. *If Conditions (C1) – (C3) hold, then*

$$\begin{pmatrix} A_n(\boldsymbol{x}) \\ B_n(\boldsymbol{x};\pi,\hat{\gamma}) \end{pmatrix} \xrightarrow{p} \begin{pmatrix} \frac{1}{\pi_0(\boldsymbol{x})} - 1 \\ \frac{1}{\pi_0(\boldsymbol{x})} - \frac{1}{\pi(\boldsymbol{x};\hat{\gamma})} \end{pmatrix}.$$

The proof is given in Appendix C and directly follows the uniform convergence of Nadaraya-Watson estimators. Note that $W_i^{\text{EEP}} = 1 + A_n(\boldsymbol{X}_i)$ and $W_i^{\text{AIPW}} = 1/\pi(\boldsymbol{X}_i;\hat{\gamma}) + B_n(\boldsymbol{X}_i;\pi,\hat{\gamma})$. Thus, Theorem 5.1 shows that both W_i^{EEP} and W_i^{AIPW} are consistent estimators of the true inverse PS, $1/\pi_0(\boldsymbol{X}_i)$. In addition, the consistency of W_i^{AIPW} does not depend on the parametric model $\pi(\cdot;\gamma)$. These results provide a clear justification of the effectiveness of the EEP and AIPW approaches, and the robustness of AIPW approach against misspecification of the parametric PS model.

5.4.6 Remarks

One challenging aspect of the EEP and AIPW approach is that when $\boldsymbol{X}_i = \boldsymbol{Z}_i$, i.e., the data are missing from all of the covariates but fully observed for the response, estimating $m(\boldsymbol{X}_i,\boldsymbol{\beta})$ requires fitting a potentially high-dimensional kernel regression that faces the "curse of dimensionality". In addition, the order of the kernel also depends on d as stated in condition (C3). High-order kernels involve local averaging with negative weights, and numerical evidences have shown that their theoretical advantages do not transfer to reasonably sized samples (Marron 1994). To alleviate the effect of high-dimension and improve finite-sample performance, we follow Xia et al. (2002) and use the refined kernel estimator

$$\tilde{\hat{m}}(\boldsymbol{X}_i,\boldsymbol{\beta}) = \frac{\sum_{j=1}^n \mathscr{K}_h(\boldsymbol{Z}_j^\top\boldsymbol{\beta} - \boldsymbol{Z}_i^\top\boldsymbol{\beta})\delta_j S(Y_j,\boldsymbol{X}_j,\boldsymbol{\beta})}{\sum_{j=1}^n \mathscr{K}_h(\boldsymbol{Z}_j^\top\boldsymbol{\beta} - \boldsymbol{Z}_i^\top\boldsymbol{\beta})\delta_j}, \text{ if } \boldsymbol{X}_i = \boldsymbol{Z}_i, \tag{5.10}$$

which simplifies (5.5) to a univariate second-order kernel regression. It is worth noting that following Theorem 5.1, the EEP and AIPW weights based on the refined kernel estimator are instead consistent for $1/P(\delta_i = 1|\boldsymbol{Z}_i^\top\boldsymbol{\beta})$ which is in general different from $1/\pi_0(\boldsymbol{Z}_i)$, but is effective of removing the selection if the data truly has a single-index structure since $\delta_i \perp Y_i \mid \boldsymbol{Z}_i^\top\boldsymbol{\beta}$. In addition, Equation (5.10) can be interpreted as locally matching the observation $(Y_i,\boldsymbol{X}_i)$ based on the prognostic score $\Psi(\boldsymbol{Z}_i) = \boldsymbol{Z}_i^\top\boldsymbol{\beta}$ which is a common practice in the missing data and causal inference literature to help with dimension reduction (Yang and Zhang 2020).

When constructing the AIPW profile pseudo-likelihood function for $\boldsymbol{\beta}$, we assume tentatively that the weights W_i^{AIPW} are nonnegative for all i. This is a limitation of the proposed method since W_i^{AIPW} can technically be negative when the estimated PS is close to zero for some subjects, rendering the likelihood invalid. A possible solution is to solve the AIPW adjusted estimating equation directly. However, solving QR estimation equation is a substantially more challenging problem than minimizing QR pseudo-likelihood since the former often requires smoothing (Chen et al. 2015; Wang et al. 2022). The estimation problem is further complicated by the fact that, the gradient $\partial g(\boldsymbol{Z}^\top\boldsymbol{\beta};\boldsymbol{\beta})/\partial\boldsymbol{\beta}$, and hence the estimating function $S(Y,\boldsymbol{X},\boldsymbol{\beta})$, does not have a closed form due to profile estimation and requires numerical approximation. To facilitate computation, we simply replace W_i^{AIPW} with W_i^{EEP} if $W_i^{\text{AIPW}}<0$. We argue that this is at least asymptotically valid since both W_i^{AIPW} and W_i^{EEP} are consistent estimators of $1/\pi_0(\boldsymbol{X}_i)$.

5.5 Implementation

We now outline the algorithm for estimating $\boldsymbol{\beta}_0$ and $g_0(\cdot)$ using the methods described in Section 5.4. In the following, we illustrate the algorithhm for the AIPW method, but the same algorithm works for IPW and EEP methods as well.

Step 0. (Initializing) Obtain initial estimator $\tilde{\boldsymbol{\beta}}$; standardize $\tilde{\boldsymbol{\beta}}$ such that $\left\|\tilde{\boldsymbol{\beta}}\right\|_2=1$ and $\tilde{\boldsymbol{\beta}}_1>0$. Estimate π_{i0} by $\pi_i(\boldsymbol{X}_i;\hat{\gamma})$, $i=1,...,n$.

Although the proposed algorithm is not sensitive to the starting value owing to its non-iterative nature, a good starting point can nonetheless benefit the nonlinear optimization algorithm used for minimizing the profile pseudo-likelihood by shortening the search path. One choice of initial estimator $\tilde{\boldsymbol{\beta}}$ of $\boldsymbol{\beta}_0$ can be obtained by minimizing the CC profile pseudo-likelihood

$$\tilde{L}_{\tau n}^{\text{CC}}(\boldsymbol{\beta})=\sum_{i=1}^{n}\delta_i\rho_\tau\left\{Y_i-\boldsymbol{B}(\boldsymbol{Z}_i^\top\boldsymbol{\beta})^\top\tilde{\boldsymbol{\theta}}^{\text{CC}}(\boldsymbol{\beta})\right\},$$

where $\tilde{\boldsymbol{\theta}}^{\text{CC}}(\boldsymbol{\beta})$ minimizes

$$L_{\tau n}^{\text{CC}}(\boldsymbol{\theta},\boldsymbol{\beta})=n^{-1}\sum_{i=1}^{n}\delta_i\rho_\tau\left\{Y_i-\boldsymbol{B}(\boldsymbol{Z}_i^\top\boldsymbol{\beta})^\top\boldsymbol{\theta}\right\}.$$

over $\boldsymbol{\theta}\in\mathbb{R}^{J_n}$.

Step 1. (Estimate β_0) The AIPW estimator $\hat{\boldsymbol{\beta}}^{\text{AIPW}}$ can be obtained by minimizing (5.8).

When the original kernel smoother defined in (5.3) is used, Equation (5.8) can be readily minimized by writing $\tilde{\boldsymbol{\theta}}(\boldsymbol{\beta})$ explicitly as a function of $\boldsymbol{\beta}$ through optimization of (5.9). When the refined kernel estimator defined in (5.10) is used, however, directly minimizing (5.8) is difficult, and thus we use an iterative reweighting procedure. Let $\hat{\boldsymbol{\beta}}^{(k)}$ be the index parameter estimates from the kth iteration, then $\hat{\boldsymbol{\beta}}^{(k+1)}$ is the solution to the following minimization problem

$$\hat{\boldsymbol{\beta}}^{(k+1)} = \arg\min_{\boldsymbol{\beta}} n^{-1} \sum_{i=1}^{n} W_i^{\text{AIPW}}(\hat{\boldsymbol{\beta}}^{(k)}) \rho_\tau \left\{ Y_i - B(\boldsymbol{Z}_i^\top \boldsymbol{\beta})^\top \tilde{\boldsymbol{\theta}}^{\text{AIPW}}(\boldsymbol{\beta}) \right\}. \tag{5.11}$$

Equation (5.11) can be called a semiparametric EM (Expectation-Minimization) algorithm, where the weighted average is the E-step and the minimization is the M-step. Given $\boldsymbol{\beta}$, Equation (5.9) corresponds to a weighted linear QR and can be solved using the *rq* function in the R package **quantreg**. Given $\tilde{\boldsymbol{\theta}}^{\text{AIPW}}(\boldsymbol{\beta})$, we solve (5.11) using the Nelder-Mead algorithm in the R package **optim**. We use a finite-difference approximation to calculate the gradient. We repeat (5.11) until $\left\| \hat{\boldsymbol{\beta}}^{(k+1)} - \hat{\boldsymbol{\beta}}^{(k)} \right\|_2^2 \leq \eta$ where η is a small positive value. We set $\eta = 10^{-6}$ for all the numerical analysis in this chapter. Finally, we standardize $\hat{\boldsymbol{\beta}}^{\text{AIPW}}$ by

$$\hat{\boldsymbol{\beta}}^{\text{AIPW}} = \text{sign}(\hat{\beta}_1^{(\text{k}+1)}) \frac{\hat{\boldsymbol{\beta}}^{(\text{k}+1)}}{\left\| \hat{\boldsymbol{\beta}}^{(\text{k}+1)} \right\|_2}.$$

Step 2. (Estimate $g_0(\cdot)$) The spline estimator of $g_0(u)$, $u \in \mathscr{U}$ is simply $\boldsymbol{B}(u)^\top \hat{\boldsymbol{\theta}}^{\text{AIPW}}(\hat{\boldsymbol{\beta}}^{\text{AIPW}})$ where $\hat{\boldsymbol{\theta}}^{\text{AIPW}}(\hat{\boldsymbol{\beta}}^{\text{AIPW}})$ minimizes

$$n^{-1} \sum_{i=1}^{n} W_i^{\text{AIPW}} \rho_\tau \left\{ Y_i - \boldsymbol{B}(\boldsymbol{Z}_i^\top \hat{\boldsymbol{\beta}}^{\text{AIPW}})^\top \boldsymbol{\theta} \right\}.$$

The support $\mathscr{U}$ is approximated by $[a_n, b_n]$ where $a_n = \min\left\{ \boldsymbol{x}_i^\top \hat{\boldsymbol{\beta}}^{\text{AIPW}} \right\}$ and $b_n = \max\left\{ \boldsymbol{x}_i^\top \hat{\boldsymbol{\beta}}^{\text{AIPW}} \right\}$.

5.5.1 Tuning parameters

The performance of the proposed methods depend on the goodness-of-fit of the spline approximation, therefore the number of internal knots needs to be optimally selected. In addition, if EEP and AIPW methods are used, choosing an optimal bandwidth is also critical for accurate estimation of the projection space. Following the results of Ma and

He (2016), we use $k_n = \lfloor n^{1/(2m+1)} \rfloor + 1$ equally spaced knots for the order m B-splines in the estimation of β_0, where $\lfloor \cdot \rfloor$ is the floor operator, so that the estimator of $g(\cdot)$ attains the optimal convergence rate. When estimating $g_0(u)$, $u \in \mathscr{U}$ given $\hat{\boldsymbol{\beta}}^{\text{AIPW}}$ (or analogously $\hat{\boldsymbol{\beta}}^{\text{IPW}}$ and $\hat{\boldsymbol{\beta}}^{\text{EEP}}$), we choose k_n to be the first local minimum of

$$\text{BIC}(k_n) = \log\left\{ L_{\tau n}^{\text{AIPW}}(\hat{\boldsymbol{\theta}}^{\text{AIPW}}, \hat{\boldsymbol{\beta}}^{\text{AIPW}}) \right\} + \frac{\log n}{2n}(k_n + m).$$

For the bandwidth parameter h, the classical optimal rate $n^{-1/(d+2l)}$ is not applicable since condition (C3) requires $nh^{2l} \to 0$. An appropriate choice is $C\, n^{-1/(d+l)}$ according to the results of Sepanski et al. (1994). When $\boldsymbol{X} = Y$, we use the rule of thumb value $h = 1.5\hat{\sigma}_Y n^{-1/3}$ where $\hat{\sigma}_Y$ is the sample standard deviation of Y. When $\boldsymbol{X} = \boldsymbol{Z}$, we simply use $h = 1.5\hat{\sigma}_{\boldsymbol{Z}^\top \boldsymbol{\beta}} n^{-1/3}$ in accordance with the refined kernel estimator.

5.6 Simulations

In this section, we study the finite performance of the proposed methods through Monte Carlo simulation. For all methods, the cubic B-spline with $m = 3$ is used to approximate the nonparametric function $g_0(\cdot)$. For IPW and AIPW methods, the PS is estimated by ordinary logistic regression. For EEP and AIPW methods, the Gaussian kernel $\mathscr{K}(u) = (\sqrt{2\pi})^{-1} \exp(-u^2/2)$ is used to estimate the conditional distribution of the missing variables. The proposed methods are compared with CC estimation. All the simulations are based on $M = 200$ replications.

5.6.1 Example 1

We consider a nonlinear model with homoscedastic errors. We generate $n = 500$ observations from the model

$$Y_i = g_0(\boldsymbol{Z}_i^\top \boldsymbol{\beta}_0) + \sigma \epsilon_i = \sin\left(\frac{\pi(\boldsymbol{Z}_i^\top \boldsymbol{\beta}_0 - A)}{C - A} \right) + \sigma \epsilon_i,\ i = 1, 2, ..., n,$$

where $\boldsymbol{\beta}_0 = \frac{1}{\sqrt{14}}(3,2,1)^\top$, $A = \sqrt{3}/2 - 1.645/\sqrt{12}$, $C = \sqrt{3}/2 + 1.645/\sqrt{12}$, and $\sigma = 0.5$. The components of $\boldsymbol{Z}_i = (\boldsymbol{Z}_{1i}, \boldsymbol{Z}_{2i}, \boldsymbol{Z}_{3i})^\top$ are independently generated from $\mathscr{U}(-1,1)$, and ϵ_i is generated from $\mathscr{N}(0,1)$. Therefore, the τth quantile of Y at a given point is $G_\tau(\boldsymbol{z}) = g_0(\boldsymbol{z}^\top \boldsymbol{\beta}_0) + \sigma \Phi^{-1}(\tau)$, where $\Phi^{-1}(\cdot)$ is the inverse CDF of standard normal distribution. We

consider two missing probability mechanisms:

$$\pi_1(y) = \frac{1}{1+\exp(-1-2y)} \text{ and } \pi_2(y) = \frac{1}{1+|y|}$$

whose corresponding missing rates MR $\approx 35\%$. It can be seen that the fully observed variable is Y and that the missing observations come from all covariates.

We evaluate the various methods with respect to the estimation accuracy of both the index parameters β_0 and the nonparametric link function $g_0(\cdot)$. Specifically, we compare the magnitude of bias (BIAS) of $\hat{\beta}_j$, standard deviation (SD) of $\hat{\beta}_j$ across M replications, and average integrated squared error (AISE) of $\hat{g}(\cdot)$ for each setting, i.e.,

$$\begin{aligned}
\text{BIAS} &= \beta_{0j} - \frac{1}{M}\sum_{i=1}^{M}\hat{\beta}_j^{(i)},\ j = 1,\ldots,p,\\
\text{SD} &= \sqrt{\frac{1}{M}\sum_{i=1}^{M}\left(\hat{\beta}_j^{(i)} - \frac{1}{M}\sum_{i=1}^{M}\hat{\beta}_j^{(i)}\right)^2},\\
\text{ARISE} &= \frac{1}{M}\sum_{i=1}^{M}\left[\frac{1}{n_{\text{grid}}}\sum_{j=1}^{n_{\text{grid}}}\left\{\hat{g}^{(i)}(u_j) - g_0(u_j)\right\}^2\right],
\end{aligned}$$

where $\hat{\beta}^{(i)}$ and $\hat{g}^{(i)}(\cdot)$ are the estimated parameters and link function from the ith replication, $u_j,\ j = 1,\ldots,n_{\text{grid}}$ are grid points of $\mathscr{U}$, and $n_{\text{grid}} = 101$.

The true and estimated curves of $g_0(u)$, $u \in [-1,1]$ for $\tau = 0.5, 0.9$ are plotted in Fig. 5.1. As expected, thhe CC estimation yields large bias since it ignores the incomplete observations even though they contain substantial information in estimating $g_0(\cdot)$. Among the proposed methods, the IPW estimator is unbiased when the logistic regression model is correct for the missing probability mechanism, and biased otherwise. In contrast, the EEP and AIPW estimators are unbiased in both cases as they incorporate additional information from the projection space. This also shows that the AIPW method is robust to misspecification of the PS model. The BIAS and SD of the estimated index parameters, and the AISE of $\hat{g}(\cdot)$ are reported in Table 5.1. Surprisingly, the CC estimators of β_0 are unbiased in both cases even though a substantial subset of observations are missing, and their efficiency is comparable to that of the proposed approaches. This suggests the estimation of the index parameters might actually be robust to missing data when single-index model is correctly specified. For the estimation of $g_0(\cdot)$, the CC estimator results in significantly larger AISE, providing

further evidence that CC estimation of the link function, and thus the conditional quantile, is subject to biasedness when covariates are MAR. Among the proposed methods, the EEP and AIPW estimators have smaller AISE compared to the IPW estimator, since the former methods directly estimate the conditional distribution of the missing observations.

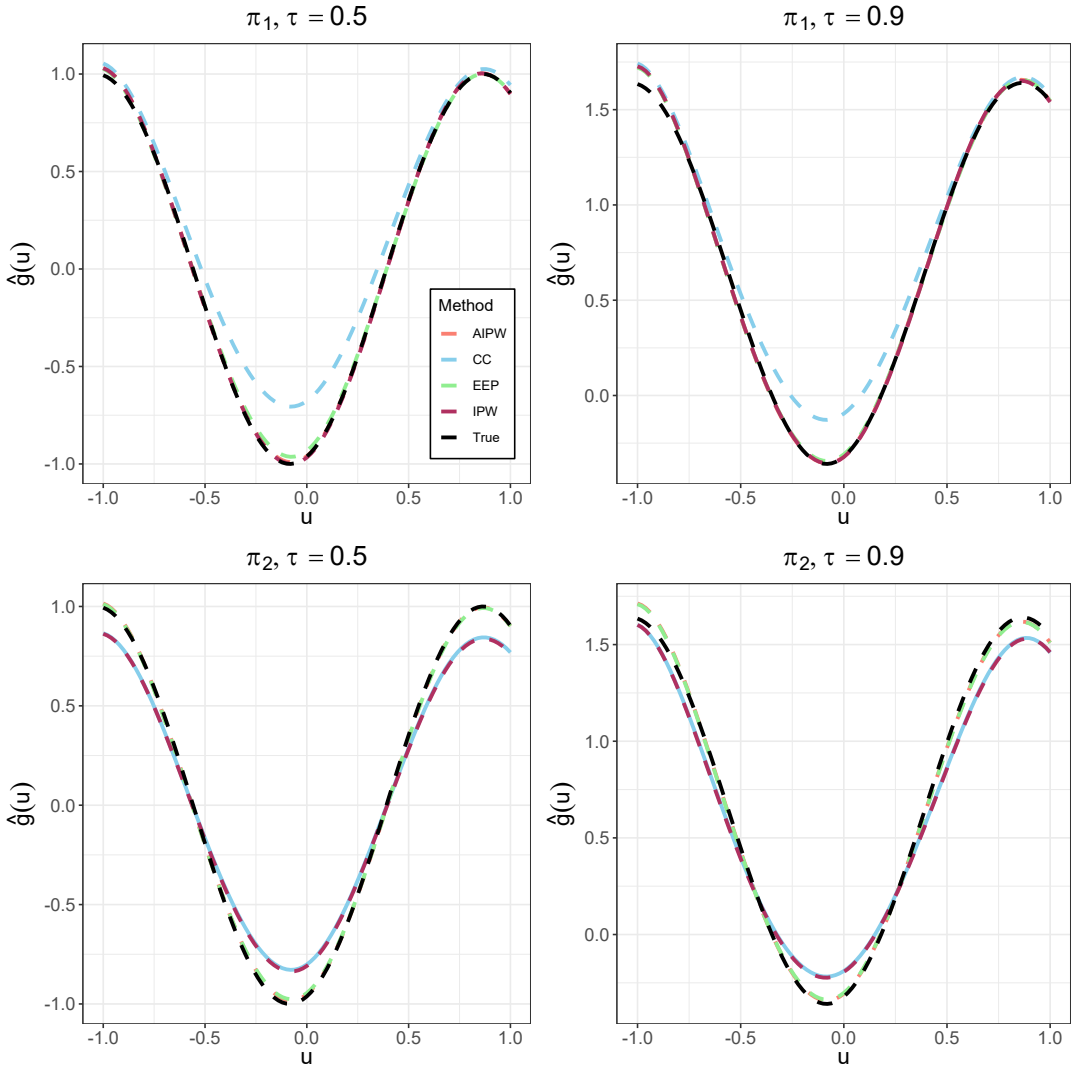


Figure 5.1: Monte Carlo study for Simulation Example 1. True and estimated curves of $g_0(u)$.

Table 5.1: Monte Carlo study for Simulation Example 1. Bias and standard deviation (SD) of the index parameter estimates, and average integrated squared error (AISE) of the link function estimate.

			BIAS$\times 10^{-2}$ (SD$\times 10^{-2}$)			
π	τ	Method	$\hat{\boldsymbol{\beta}}_{01}$	$\hat{\boldsymbol{\beta}}_{02}$	$\hat{\boldsymbol{\beta}}_{03}$	AISE$\times 10^{-2}$
π_1	0.5	CC	-0.03 (1.80)	-0.32 (2.53)	-0.38 (3.03)	3.41
		IPW	-0.06 (1.87)	-0.23 (2.56)	-0.23 (3.37)	0.87
		EEP	-0.10 (1.82)	-0.15 (2.46)	-0.23 (3.23)	**0.76**
		AIPW	-0.09 (1.86)	-0.17 (2.56)	0.23 (3.26)	**0.76**
	0.9	CC	-0.02 (2.45)	-0.14 (3.25)	-0.33 (4.28)	2.84
		IPW	0.03 (2.06)	-0.27 (2.86)	0.33 (3.19)	1.19
		EEP	0.01 (2.03)	-0.24 (2.85)	0.01 (3.36)	**1.14**
		AIPW	0.01 (2.06)	-0.23 (2.84)	-0.01 (3.37)	1.17
π_2	0.5	CC	0.05 (1.78)	-0.32 (2.27)	0.16 (3.03)	2.10
		IPW	0.06 (1.79)	-0.33 (2.29)	0.15 (3.01)	2.08
		EEP	-0.02 (1.67)	-0.15 (2.19)	0.07 (2.92)	**0.77**
		AIPW	-0.04 (1.67)	-0.13 (2.20)	0.08 (2.88)	**0.77**
	0.9	CC	-0.13 (2.40)	-0.03 (3.59)	-0.30 (4.59)	2.41
		IPW	-0.14 (2.37)	-0.01 (3.53)	-0.26 (4.48)	2.41
		EEP	-0.17 (2.36)	0.00 (3.40)	-0.26 (4.97)	**1.80**
		AIPW	-0.15 (2.36)	-0.03 (3.38)	-0.25 (4.90)	1.83

Note: best-performing estimators with the smallest AISE are in bold fonts.

5.6.2 Example 2

We consider a nonlinear model with heteroscedastic errors. We generate $n = 300$ observations from the model

$$Y_i = g_0(\boldsymbol{Z}_i^\top \boldsymbol{\beta}_0) + \sigma(\boldsymbol{Z}_i^\top \boldsymbol{\beta}_0)\epsilon_i = 5\cos\left(\pi \boldsymbol{Z}_i^\top \boldsymbol{\beta}_0\right) + \sqrt{1 + \left(\boldsymbol{Z}_i^\top \boldsymbol{\beta}_0\right)^2}\epsilon_i, i = 1, ..., n,$$

where $\boldsymbol{\beta}_0 = \frac{1}{\sqrt{6}}(1, -2, 1)^\top$. The components of $\boldsymbol{Z}_i = (\boldsymbol{Z}_{1i}, \boldsymbol{Z}_{2i}, \boldsymbol{Z}_{3i})^\top$ are independently generated from $\mathscr{U}(-1, 1)$, and ϵ_i is generated from $\mathscr{N}(0, 1)$. Therefore, the τth quantile at Y at a given point is $G_\tau(\boldsymbol{z}) = g_0(\boldsymbol{z}^\top \boldsymbol{\beta}_0) + \sigma(\boldsymbol{z}^\top \boldsymbol{\beta}_0)\Phi^{-1}(\tau)$. We consider two missing probability mechanisms:

$$\pi_1(\boldsymbol{z}) = \frac{1}{1 + \exp(-0.986 - 1.5z_2)} \text{ and } \pi_2(\boldsymbol{z}) = \frac{1}{1 + (z_1 + z_3)^2},$$

whose corresponding missing rates MR $\approx 40\%$. It can be seen that all covariates are observed and the missing observations come from the response only. In this case, it is known that the CC estimator the conditional quantile is unbiased, and thus we expect that the proposed methods will not yield significant improvements.

The true and estimated curves of $g_0(u)$, $u \in [-1, 1]$ for $\tau = 0.5, 0.75$ are plotted in Fig. 5.2. The results show that all methods yield unbiased estimates of the link function. The BIAS and SD of the estimated index parameters, and the AISE of $\hat{g}(\cdot)$ are reported in Table 5.2. When $\pi = \pi_1$, the index parameter estimates based on CC and EEP methods have significantly larger BIAS (and larger SD as well when $\tau = 0.5$) than that of IPW and AIPW approaches. After examining the results, we found that both CC and EEP methods returned many outlier estimates which substantially affect their overall performance. This suggests that when the PS model is correctly specified, the AIPW estimator is robust against poorly estimated conditional distribution of missing variables. This phenomena is also reflected by the AISE, for which IPW and AIPW estimators have smaller values compared to CC and EEP estimators. When $\pi = \pi_2$, such that the logistic regression model is misspecified, the EEP and AIPW methods yield smaller AISE compared to CC and IPW methods. The AIPW is less efficient than the EEP $\tau = 0.75$ since the poorly estimated PS introduced additional variance.

Table 5.2: Monte Carlo study for Simulation Example 2. Bias and standard deviation (SD) of the index parameter estimates, and average integrated squared error (AISE) of the link function estimate.

			BIAS$\times 10^{-2}$ (SD$\times 10^{-1}$)			
π	τ	Method	$\hat{\boldsymbol{\beta}}_1$	$\hat{\boldsymbol{\beta}}_2$	$\hat{\boldsymbol{\beta}}_3$	AISE
π_1	0.5	CC	-0.05 (0.15)	0.31 (0.51)	0.17 (0.37)	0.23
		IPW	-0.04 (0.15)	-0.01 (0.10)	-0.05 (0.15)	**0.12**
		EEP	-0.05 (0.15)	0.32 (0.51)	0.18 (0.37)	0.24
		AIPW	-0.07 (0.15)	-0.03 (0.10)	-0.05 (0.15)	**0.12**
	0.75	CC	0.68 (0.39)	1.47 (1.13)	0.30 (0.38)	0.47
		IPW	0.53 (0.37)	1.05 (0.97)	0.14 (0.31)	0.37
		EEP	0.67 (0.40)	1.49 (1.14)	0.33 (0.36)	0.47
		AIPW	0.35 (0.22)	0.70 (0.82)	0.07 (0.28)	**0.36**
π_2	0.5	CC	0.97 (0.63)	2.74 (1.70)	0.03 (0.54)	0.74
		IPW	0.96 (0.63)	2.75 (1.71)	0.05 (0.54)	0.74
		EEP	0.99 (0.62)	2.76 (1.69)	0.09 (0.54)	**0.72**
		AIPW	0.98 (0.65)	2.72 (1.67)	0.09 (0.53)	**0.72**
	0.75	CC	0.65 (0.58)	3.58 (1.66)	1.60 (0.88)	1.18
		IPW	0.50 (0.58)	3.26 (1.68)	1.23 (0.81)	1.10
		EEP	0.55 (0.50)	1.50 (1.10)	0.37 (0.47)	**0.57**
		AIPW	0.76 (0.62)	1.90 (1.22)	0.39 (0.51)	0.69

Note: best-performing estimators with the smallest AISE are in bold fonts.

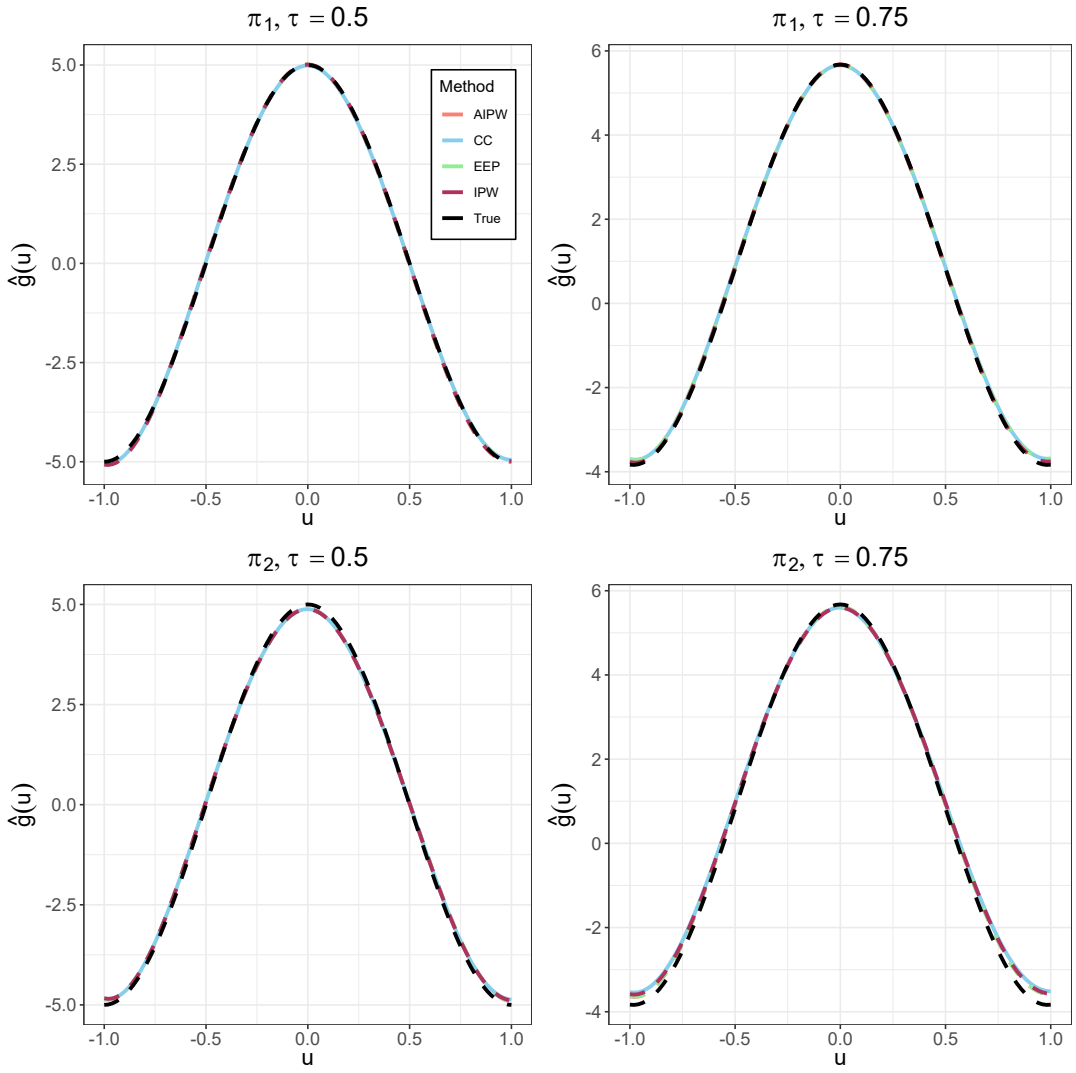


Figure 5.2: Monte Carlo study for Simulation Example 2. True and estimated curves of $g_0(u)$.

5.7 Data application

As an illustration, we apply the proposed methods to the body fat data set (Johnson 1996), which is available from the R package **mfp**. The data contain 252 observations on 17 variables. The response variable is the estimated Percent body fat (using Brozek's equation: 457/Density−414.2) determined by underwater weighing, and the covariates are various body circumference measurements. This data set has been analyzed previously in the single-index model context. Examples include Liu and Yang (2017) and Li et al. (2017) who

considered composite QR with variable selection, and Liu et al. (2019) who considered composite QR with covariates MAR. These works all identified age (years), abdomen (cm) and wrist (cm) as significant variables for single-index regression on percentage body fat. Based on their results, we set the response $Y = \log(\text{Percent body fat})$ and covariates $Z_1 = \text{age}$, $Z_2 = \text{abdomen}$, $Z_3 = \text{wrist}$. Our main interest is to study the relationship between the three measurements and the median ($\tau = 0.5$) of the log-transformed percentage body fat, which can be modeled using the following single-index QR

$$Q_{0.5}(Y_i|\boldsymbol{Z}_i = \boldsymbol{z}_i) = g_0(\beta_{01}\text{age}_i + \beta_{02}\text{abdomen}_i + \beta_{03}\text{wrist}_i), i = 1, ..., n.$$

We exclude an observation with estimated percentage body fat of 0 (case 182) and an observation with extreme abdominal circumference (case 39). We then standardize all of the covariates to have mean 0 and standard deviation 1.

It is worth noting that the original data set does not contain missing values. To demonstrate the advantage of the proposed methods in handling MAR observations, we use a pre-specified missing data mechanism to determine whether each case should have missing data or not. We then use both complete and missing data to model this data set, where the model fitted using the complete data serves as a benchmark to evaluate the effectiveness of the proposed methods. For the case of complete data, we approximate the nonparametric function $g_0(\cdot)$ using quadratic B-spline with $m = 2$. The estimated curve of $g_0(\cdot)$, along with the scatterplots of the observed response Y_i and estimated single-index $\boldsymbol{Z}_i^\top \hat{\boldsymbol{\beta}}$ are plotted in Fig. 5.3. The results show that $g(\cdot)$ is clearly nonlinear and overall monotonically increasing. The estimated index parameters are $\hat{\boldsymbol{\beta}} = (0.1873, 0.9507, -0.2472)^\top$ with bootstrap standard errors $\text{se}(\hat{\boldsymbol{\beta}}) = (0.0252, 0.0070, 0.0235)^\top$ obatined using the approach outlined in Wu et al. (2010). This suggests that age and larger abdominal circumference are associated with higher percentage body fat, and larger wrist circumference is associated with lower percentage body fat. These findings coincide with those reported in Liu et al. (2019).

For the case of missing data, we randomly delete a portion of the observed covariates using two missing probability mechanisms:

$$\pi_1(y) = \frac{1}{1+\exp(2y-6.8)} \text{ and } \pi_2(y) = \frac{1}{1+\exp(-2.2y+5.8)}$$

whose corresponding missing rates MR $\approx 30\%$ and 50%, respectively. We generate $M = 200$ missing replicates from the original data set to estimate $g_0(\cdot)$ and $\boldsymbol{\beta}_0$ using the proposed

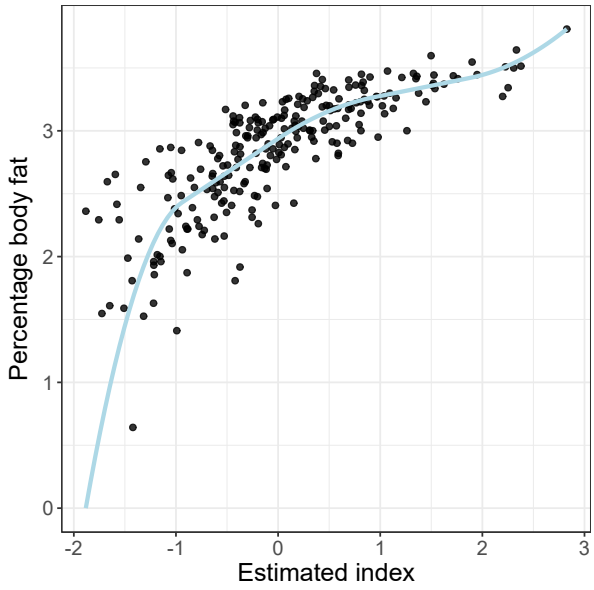


Figure 5.3: Body fat analysis. Estimated curve of the link function $g_0(\cdot)$. The scatterplots are observed of response Y_i and estimated single-index $\boldsymbol{Z}_i^\top \hat{\boldsymbol{\beta}}$.

methods. The estimated $g_0(u)$, $u \in [-1.4, 2.2]$ and its 95% Monte Carlo confidence intervals (CI) using the full data, CC estimation, and the proposed methods are plotted in Fig. 5.4. When the missing observations are generated by π_1, the CC estimation is associated with large variance due to multiple outlier estimates, whereas the proposed methods significantly improved the estimation efficiency by incorporating additional information from the fully observed variables. When the missing observations are generated by π_2, the CC estimation yields narrow CI but is inconsistent for u close to -0.5, whereas the proposed methods yield consistent estimation with small variance. The average (Estimate) and standard deviation (SD) of the estimated $\boldsymbol{\beta}_0$ over M replications are reported in Table 5.3. The results show that, compared to the CC estimation, the estimated parameters based on the proposed methods are closer to the ones obtained from the complete data. In particular, the proposed methods significantly improves the estimation efficiency of $\hat{\boldsymbol{\beta}}_2$.

5.8 Discussion

In this chapter, we considered conditional quantile estimation given multiple covariates using single-index model when either the covariates or the response is MAR. To incorporate

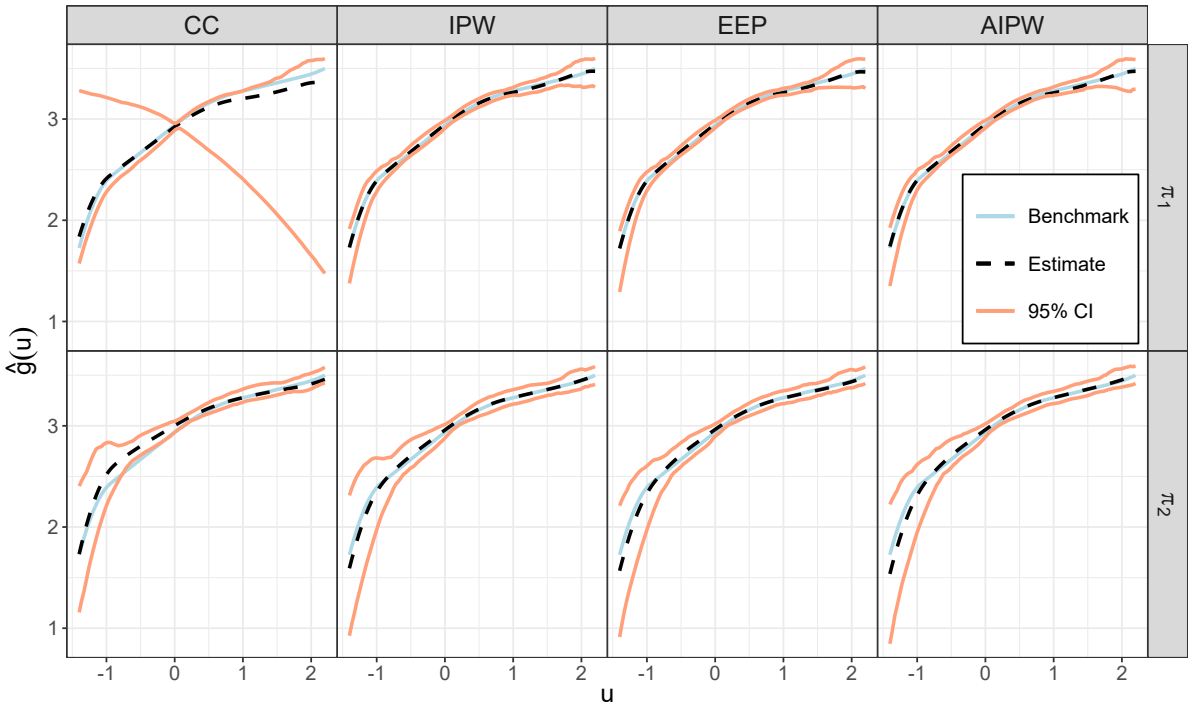


Figure 5.4: Body fat analysis. Estimated curve of the link function $g_0(\cdot)$ when covariates are MAR.

Table 5.3: Body fat analysis. Estimated single-index coefficients and their Monte Carlo standard deviation (SD).

		Estimate (SD)		
π	Method	$\hat{\beta}_1$	$\hat{\beta}_2$	$\hat{\beta}_3$
	Full	0.1873 (—)	0.9507 (—)	-0.2472 (—)
π_1	CC	0.1847 (0.0360)	0.8363 (0.4639)	-0.2114 (0.0806)
	IPW	0.1794 (0.0347)	0.9433 (0.1381)	-0.2331 (0.0588)
	EEP	0.1801 (0.0357)	0.9431 (0.1387)	-0.2362 (0.0448)
	AIPW	0.1795 (0.0350)	0.9340 (0.1927)	-0.2312 (0.0630)
π_2	CC	0.1922 (0.0665)	0.8903 (0.2707)	-0.2832 (0.1141)
	IPW	0.2038 (0.0854)	0.9155 (0.1430)	-0.2769 (0.1270)
	EEP	0.2032 (0.0874)	0.9054 (0.1941)	-0.2796 (0.1259)
	AIPW	0.2030 (0.0933)	0.9144 (0.1433)	-0.2815 (0.1120)

additional information from the fully observed variables, we proposed a class of weighted pseudo-likelihoods that includes but goes beyond IPW estimation. Using spline approximation and profile principle, we construct weighted QR estimators for both the index parameters and link function, which can be computed using an efficient algorithm similar to that of Ma and He (2016). We also show that the EEP and AIPW approaches are essentially a nonparametric IPW approach in the asymptotic sense, justifying their advantage over the parametric IPW approach from a novel perspective. Results from simulation and real-data application provide empirical evidence that the proposed methods lead to smaller estimation bias when the covariates are missing, and more numerically stable estimation algorithm in general.

Although the focus of this chapter is on missing covariates or response, the proposed methods can be easily extended to handle the general MAR setting. In fact, the IPW approach has already been studied in such setting (Liu and Yang 2017; Liang et al. 2021). For EEP and AIPW approaches, some thoughts need to be given on how to appropriately formulate the kernel estimator when a subset of $\boldsymbol{Z}$, or a subset of $\boldsymbol{Z}$ in addition to Y, is missing while avoiding the “curse of dimensionality". One possible solution is to extend th refined kernel estimator given in Section 5.4 to the case of partial covariates. Let $\boldsymbol{Z}^o$ be the fully observed covariates, then instead of directly conditioning on $\boldsymbol{Z}^o$ one can condition on $\boldsymbol{Z}^{o\top}\boldsymbol{\beta}^o$ where $\boldsymbol{\beta}^o$ is the subvector of single-index coefficients for $\boldsymbol{Z}^o$.

As mentioned in Ma and He (2016), the advantage of profile estimation is the availability of a single objective function as a function of $\boldsymbol{\beta}$. This allows straightforward implementation of sparsity-induced penalties such as LASSO (Tibshirani 1996) or SCAD (Fan and Li 2001). Therefore, a possible future research direction is to study the statistical properties of the proposed estimators under such penalties.

REFERENCES

Abrahamowicz, M., Clampl, A., and Ramsay, J. O. (1992). Nonparametric density estimation for censored survival data: Regression-spline approach. *Canadian Journal of Statistics*, 20(2):171–185.

Abrevaya, J. (2001). The effects of demographics and maternal behavior on the distribution of birth outcomes. *Empirical Economics*, 26:247–257.

Abrevaya, J., Hsu, Y.-C., and Lieli, R. P. (2015). Estimating conditional average treatment effects. *Journal of Business & Economic Statistics*, 33(4):485–505.

Apley, D. (2018). *ALEPlot: Accumulated Local Effects (ALE) Plots and Partial Dependence (PD) Plots*. R package version 1.1.

Apley, D. W. and Zhu, J. (2020). Visualizing the effects of predictor variables in black box supervised learning models. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 82:1059–1086.

Beatson, R. (1982). Restricted range approximation by splines and variational inequalities. *SIAM Journal on Numerical Analysis*, 19(2):372–380.

Benoit, D. F. and Van den Poel, D. (2017). bayesQR: A bayesian approach to quantile regression. *Journal of Statistical Software*, 76(7):1–32.

Betancourt, M. (2017). A conceptual introduction to Hamiltonian Monte Carlo. *arXiv preprint arXiv:1701.02434*.

Betancourt, M. and Girolami, M. (2015). Hamiltonian Monte Carlo for hierarchical models. *Current Trends in Bayesian Methodology with Applications*, 79:2–4.

Bishop, C. M. et al. (1995). *Neural networks for pattern recognition*. Oxford university press.

Bondell, H. D., Reich, B. J., and Wang, H. (2010). Noncrossing quantile regression curve estimation. *Biometrika*, 97:825–838.

Cannon, A. J. (2018). Non-crossing nonlinear regression quantiles by monotone composite quantile regression neural network, with application to rainfall extremes. *Stochastic Environmental Research and Risk Assessment*, 32:3207–3225.

Chen, X., Wan, A. T., and Zhou, Y. (2015). Efficient Quantile Regression Analysis with Missing Observations. *Journal of the American Statistical Association*, 110(510):723–741.

Chernozhukov, V., Fernández-Val, I., and Galichon, A. (2010). Quantile and probability curves without crossing. *Econometrica*, 78:1093–1125.

Chernozhukov, V., Fernández-Val, I., and Melly, B. (2013). Inference on counterfactual distributions. *Econometrica*, 81(6):2205–2268.

Chipman, H. A., George, E. I., and McCulloch, R. E. (2010). BART: Bayesian additive regression trees. *The Annals of Applied Statistics*, 4(1):266–298.

Christou, E. and Akritas, M. G. (2016). Single Index Quantile Regression for Heteroscedastic Data. *Journal of Multivariate Analysis*, 150:169–182.

Chui, C., Smith, P., and Ward, J. (1980). Degree of L_p Approximation by Monotone Splines. *SIAM Journal on Mathematical Analysis*, 11(3):436–447.

Cui, T., Havulinna, A., Marttinen, P., and Kaski, S. (2021). Informative Bayesian neural network priors for weak signals. *Bayesian Analysis*, 1(1):1–31.

Das, P. and Ghosal, S. (2018). Bayesian non-parametric simultaneous quantile regression for complete and grid data. *Computational Statistics & Data Analysis*, 127:172–186.

Dette, H. and Volgushev, S. (2008). Non-Crossing Non-Parametric Estimates of Quantile Curves. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 70(3):609–627.

DiNardo, J., Fortin, N. M., and Lemieux, T. (1996). Labor Market Institutions and the Distribution of Wages, 1973-1992: A Semiparametric Approach. *Econometrica*, 64(5):1001–1044.

Doksum, K. (1974). Empirical probability plots and statistical inference for nonlinear models in the two-sample case. *The Annals of Statistics*, pages 267–277.

Donald, S. G. and Hsu, Y.-C. (2014). Estimation and inference for distribution functions and quantile functions in treatment effect models. *Journal of Econometrics*, 178:383–397.

Eddelbuettel, D., Francois, R., Allaire, J., Ushey, K., Kou, Q., Russell, N., Ucar, I., Bates, D., and Chambers, J. (2022a). *Rcpp: Seamless R and C++ Integration*. R package version 1.0.8.

Eddelbuettel, D., Francois, R., Bates, D., Ni, B., and Sanderson, C. (2022b). *RcppArmadillo: 'Rcpp' Integration for the 'Armadillo' Templated Linear Algebra Library*. R package version 0.10.8.1.0.

Falbel, D. and Luraschi, J. (2022). *torch: Tensors and Neural Networks with 'GPU' Acceleration*. R package version 0.7.2.

Fan, J. and Gijbels, I. (1992). Variable Bandwidth and Local Linear Regression Smoothers. *The Annals of Statistics*, pages 2008–2036.

Fan, J. and Li, R. (2001). Variable selection via nonconcave penalized likelihood and its oracle properties. *Journal of the American statistical Association*, 96(456):1348–1360.

Fasiolo, M., Wood, S. N., Zaffran, M., Nedellec, R., and Goude, Y. (2021). qgam: Bayesian nonparametric quantile regression modeling in R. *Journal of Statistical Software*, 100(9):1–31.

Firpo, S. (2007). Efficient semiparametric estimation of quantile treatment effects. *Econometrica*, 75(1):259–276.

Fortuin, V., Garriga-Alonso, A., Wenzel, F., Rätsch, G., Turner, R., van der Wilk, M., and Aitchison, L. (2021). Bayesian neural network priors revisited. *arXiv preprint arXiv:2102.06571*.

Friedman, J. H. (2001). Greedy function approximation: a gradient boosting machine. *Annals of statistics*, pages 1189–1232.

Frölich, M. and Melly, B. (2013). Unconditional quantile treatment effects under endogeneity. *Journal of Business & Economic Statistics*, 31(3):346–357.

Galarza, C. E., Benites, L., Bourguignon, M., and Lachos, V. H. (2022). *lqr: Robust Linear Quantile Regression*. R package version 4.1.

Gelman, A. et al. (2006). Prior distributions for variance parameters in hierarchical models. *Bayesian analysis*, 1(3):515–534.

Geman, S. and Geman, D. (1984). Stochastic relaxation, Gibbs distributions, and the Bayesian restoration of images. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 6(6):721–741.

Ghosh, S., Yao, J., and Doshi-Velez, F. (2019). Model Selection in Bayesian Neural Networks via Horseshoe Priors. *Journal of Machine Learning Research*, 20(182):1–46.

Greenwell, B. M., Boehmke, B. C., and McCarthy, A. J. (2018). A simple and effective model-based variable importance measure. *arXiv preprint arXiv:1805.04755*.

Guo, J., Gabry, J., Goodrich, B., and Weber, S. (2021). *rstan: R Interface to Stan*. R package version 2.21.3.

Hardle, W., Hall, P., and Ichimura, H. (1993). Optimal Smoothing in Single-Index Models. *The Annals of Statistics*, 21(1):157–178.

He, X. (1997). Quantile curves without crossing. *The American Statistician*, 51(2):186–192.

Hoffman, M. D. and Gelman, A. (2014). The No-U-Turn sampler: adaptively setting path lengths in Hamiltonian Monte Carlo. *Journal of Machine Learning Research*, 15:1593–1623.

Holmes, M. P., Gray, A. G., and Isbell, C. L. (2012). Fast nonparametric conditional density estimation. *arXiv preprint arXiv:1206.5278*.

Hornik, K., Stinchcombe, M., White, H., et al. (1989). Multilayer feedforward networks are universal approximators. *Neural Networks*, 2:359–366.

Hu, Y., Gramacy, R. B., and Lian, H. (2013). Bayesian quantile regression for single-index models. *Statistics and Computing*, 23(4):437–454.

Hu, Z., Follmann, D. A., and Qin, J. (2012). Semiparametric double balancing score estimation for incomplete data with ignorable missingness. *Journal of the American Statistical Association*, 107(497):247–257.

Huang, M.-Y. and Yang, S. (2020). Robust inference of conditional average treatment effects using dimension reduction. *arXiv preprint arXiv:2008.13137*.

Ichimura, H. (1993). Semiparametric Least Squares (SLS) and Weighted SLS Estimation of Single-Index Models. *Journal of econometrics*, 58(1-2):71–120.

Imai, K., King, G., and Stuart, E. A. (2008). Misunderstandings between experimentalists and observationalists about causal inference. *Journal of the Royal Statistical Society: Series A (Statistics in Society)*, 171(2):481–502.

Imbens, G. W. and Rubin, D. B. (2015). *Causal inference in statistics, social, and biomedical sciences*. Cambridge University Press.

Imbens, G. W. and Wooldridge, J. M. (2009). Recent developments in the econometrics of program evaluation. *Journal of Economic Literature*, 47(1):5–86.

Izbicki, R. and Lee, A. B. (2016). Nonparametric conditional density estimation in a high-dimensional regression setting. *Journal of Computational and Graphical Statistics*, 25(4):1297–1316.

Jagger, T. H. and Elsner, J. B. (2009). Modeling tropical cyclone intensity with quantile regression. *International Journal of Climatology*, 29(10):1351–1361.

Jiang, R. and Yu, K. (2021). No-Crossing Single-Index Quantile Regression Curve Estimation. *Journal of Business & Economic Statistics*, (just-accepted):1–35.

Johnson, R. W. (1996). Fitting percentage of body fat to simple body measurements. *Journal of Statistics Education*, 4(1).

Kennedy, E. H., Balakrishnan, S., and Wasserman, L. (2021). Semiparametric counterfactual density estimation. *arXiv preprint arXiv:2102.12034*.

Kim, K., Kim, J., and Kennedy, E. H. (2018). Causal effects based on distributional distances. *arXiv preprint arXiv:1806.02935*.

Kim, T., Fakoor, R., Mueller, J., Smola, A. J., and Tibshirani, R. J. (2021). Deep Quantile Aggregation. *arXiv preprint arXiv:2103.00083*.

Kingma, D. P. and Ba, J. (2014). Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*.

Koenker, R. (2005). *Quantile Regression*. Cambridge University Press.

Koenker, R. (2022). *quantreg: Quantile Regression*. R package version 5.88.

Koenker, R. and Bassett Jr, G. (1978). Regression quantiles. *Econometrica*, 46:33–50.

Kong, E. and Xia, Y. (2012). A Single-Index Quantile Regression Model and Its Estimation. *Econometric Theory*, 28(4):730–768.

Kuhn, M. (2022). *caret: Classification and Regression Training*. R package version 6.0-91.

Lehmann, E. L. and D'Abrera, H. J. (1975). *Nonparametrics: Statistical Methods Based on Ranks*. Holden-Day.

Li, J., Li, Y., and Zhang, R. (2017). B spline variable selection for the single index models. *Statistical Papers*, 58(3):691–706.

Li, R., Bondell, H. D., and Reich, B. J. (2021). Deep distribution regression. *In press, Computational Statistics and Data Analysis*.

Liang, H.-Y., Wang, B.-H., and Shen, Y. (2021). Quantile regression of partially linear single-index model with missing observations. *Statistics*, 55(1):1–17.

Lipsitz, M., Belloni, A., Chernozhukov, V., and Fernandez-Val, I. (2016). *quantreg.nonpar: Nonparametric Series Quantile Regression*. R package version 1.0.

Little, R. J. and Rubin, D. B. (2019). *Statistical Analysis with Missing Data*. New York: Wiley.

Liu, C.-S. and Liang, H.-Y. (2022). Bayesian analysis in single-index quantile regression with missing observation. *Communications in Statistics-Theory and Methods*, pages 1–29.

Liu, H. and Yang, H. (2017). Estimation and variable selection in single-index composite quantile regression. *Communications in Statistics-Simulation and Computation*, 46(9):7022–7039.

Liu, H., Yang, H., and Peng, C. (2019). Weighted composite quantile regression for single index model with missing covariates at random. *Computational Statistics*, 34(4):1711–1740.

Liu, Y. and Wu, Y. (2009). Stepwise multiple quantile regression estimation using non-crossing constraints. *Statistics and Its Interface*, 2:299–310.

Liu, Y. and Wu, Y. (2011). Simultaneous multiple non-crossing quantile regression estimation using kernel constraints. *Journal of Nonparametric Statistics*, 23:415–437.

Lu, M., Sadiq, S., Feaster, D. J., and Ishwaran, H. (2018). Estimating individual treatment effect in observational data using random forest methods. *Journal of Computational and Graphical Statistics*, 27(1):209–219.

Ma, S. and He, X. (2016). Inference for Single-Index Quantile Regression Models with Profile Optimization. *The Annals of Statistics*, 44(3):1234–1268.

Ma, Y. and Zhu, L. (2013). A review on dimension reduction. *International Statistical Review*, 81(1):134–150.

Mack, Y.-p. and Silverman, B. W. (1982). Weak and strong uniform consistency of kernel regression estimates. *Zeitschrift für Wahrscheinlichkeitstheorie und verwandte Gebiete*, 61(3):405–415.

MacKay, D. J. (1992). A practical Bayesian framework for backpropagation networks. *Neural computation*, 4(3):448–472.

Maidman, A. (2017). *plaqr: Partially Linear Additive Quantile Regression*. R package version 2.0.

Marron, J. (1994). Visual understanding of higher-order kernels. *Journal of Computational and Graphical Statistics*, 3(4):447–458.

Matthews, A. G. d. G., Rowland, M., Hron, J., Turner, R. E., and Ghahramani, Z. (2018). Gaussian Process Behaviour in Wide Deep Neural Networks. *arXiv preprint arXiv:1804.11271*.

Metropolis, N., Rosenbluth, A. W., Rosenbluth, M. N., Teller, A. H., and Teller, E. (1953). Equation of state calculations by fast computing machines. *The Journal of Chemical Physics*, 21(6):1087–1092.

Microsoft and Weston, S. (2022). *foreach: Provides Foreach Looping Construct*. R package version 1.5.2.

Monnahan, C. C. and Kristensen, K. (2018). No-U-turn sampling for fast Bayesian inference in ADMB and TMB: Introducing the adnuts and tmbstan R packages. *PLoS ONE*, 13:e0197954.

Moon, S. J., Jeon, J.-J., Lee, J. S. H., and Kim, Y. (2021). Learning Multiple Quantiles with Neural Networks. *Journal of Computational and Graphical Statistics*, 30(4):1238–1248.

Muggeo, V. M., Sciandra, M., Tomasello, A., and Calvo, S. (2013). Estimating growth charts via nonparametric quantile regression: a practical framework with application in ecology. *Environmental and Ecological Statistics*, 20:519–531.

National Center for Health Statistics (2019). 2019 Natality. data retrieved from Centers for Disease Control and Prevention, https://ftp.cdc.gov/pub/Health_Statistics/NCHS/Datasets/DVS/natality/.

Neal, R. M. (1993). Bayesian learning via stochastic dynamics. In *Advances in neural information processing systems*, pages 475–482.

Neal, R. M. (1996). *Bayesian Learning for Neural Networks*. Springer.

Neal, R. M. (2011). MCMC using Hamiltonian dynamics. In Brooks, S., Gelman, A., Jones, G. L., and Meng, X. L., editors, *Handbook of Markov Chain Monte Carlo*, chapter 5, pages 113–162. Chapman & Hall/CRC.

Neal, R. M. (2012). *Bayesian learning for neural networks*, volume 118. Springer Science & Business Media.

Ngwira, A. and Stanley, C. C. (2015). Determinants of low birth weight in Malawi: Bayesian geo-additive modelling. *PloS one*, 10(6):e0130057.

Paszke, A., Gross, S., Massa, F., Lerer, A., Bradbury, J., Chanan, G., Killeen, T., Lin, Z., Gimelshein, N., Antiga, L., Desmaison, A., Köpf, A., Yang, E., DeVito, Z., Raison, M., Tejani, A., Chilamkurthy, S., Steiner, B., Fang, L., Bai, J., and Chintala, S. (2019). Pytorch: An imperative style, high-performance deep learning library.

Ramsay, J. O. (1988). Monotone regression splines in action. *Statistical science*, pages 425–441.

Reich, B. J. and Smith, L. B. (2013). Bayesian quantile regression for censored data. *Biometrics*, 69:651–660.

Ribeiro, M. T., Singh, S., and Guestrin, C. (2016). Model-agnostic interpretability of machine learning. *arXiv preprint arXiv:1606.05386*.

Robins, J. M. and Rotnitzky, A. (2001). Comment on the Bickel and Kwon article,"Inference for semiparametric models: Some questions and an answer". *Statistica Sinica*, 11(4):920–936.

Rodrigues, T. and Fan, Y. (2017). Regression adjustment for noncrossing Bayesian quantile regression. *Journal of Computational and Graphical Statistics*, 26:275–284.

Rothe, C. (2010). Nonparametric estimation of distributional policy effects. *Journal of Econometrics*, 155(1):56–70.

Rubin, D. B. (1978). Bayesian inference for causal effects: The role of randomization. *The Annals of Statistics*, pages 34–58.

Rubin, D. B. (1981). The Bayesian bootstrap. *The Annals of Statistics*, pages 130–134.

Salvatier, J., Wiecki, T. V., and Fonnesbeck, C. (2016). Probabilistic programming in Python using PyMC3. *PeerJ Computer Science*, 2:e55.

Sepanski, J., Knickerbocker, R., and Carroll, R. (1994). A Semiparametric Correction for Attenuation. *Journal of the American Statistical Association*, 89(428):1366–1373.

Sherwood, B., Wang, L., and Zhou, X.-H. (2013). Weighted quantile regression for analyzing health care cost data with missing covariates. *Statistics in Medicine*, 32(28):4967–4979.

Smith, L. B., Reich, B. J., Herring, A. H., Langlois, P. H., and Fuentes, M. (2015). Multilevel quantile function modeling with application to birth outcomes. *Biometrics*, 71(2):508–519.

Stan Development Team (2019). Stan Modeling Language Users Guide and Reference Manual.

Stan Development Team (2022). *Stan Modeling Language Users Guide and Reference Manual*. version 2.29.

Stute, W. (1982). The oscillation behavior of empirical processes. *The annals of Probability*, pages 86–107.

Sun, S., Moodie, E. E., and Nešlehová, J. G. (2021). Causal inference for quantile treatment effects. *Environmetrics*, 32(4):e2668.

Taddy, M. A. and Kottas, A. (2010). A Bayesian Nonparametric Approach to Inference for Quantile Regression. *Journal of Business & Economic Statistics*, 28(3):357–369.

Takeuchi, I., Le, Q. V., Sears, T. D., and Smola, A. J. (2006). Nonparametric quantile estimation. *Journal of Machine Learning Research*, 7:1231–1264.

Tibshirani, R. (1996). Regression shrinkage and selection via the lasso. *Journal of the Royal Statistical Society: Series B (Methodological)*, 58(1):267–288.

Tokdar, S. and Cunningham, E. (2019). *qrjoint: Joint Estimation in Linear Quantile Regression*. R package version 2.0-3.

Tokdar, S. T., Kadane, J. B., et al. (2012). Simultaneous linear quantile regression: a semiparametric Bayesian approach. *Bayesian Analysis*, 7:51–72.

Vansteelandt, S. and Daniel, R. M. (2014). On regression adjustment for the propensity score. *Statistics in medicine*, 33(23):4053–4072.

Vehtari, A., Gabry, J., Magnusson, M., Yao, Y., Bürkner, P.-C., Paananen, T., and Gelman, A. (2020). *loo: Efficient Leave-One-Out Cross-Validation and WAIC for Bayesian Models*. R package version 2.4.1.

Vehtari, A., Gelman, A., and Gabry, J. (2017). Practical Bayesian model evaluation using leave-one-out cross-validation and WAIC. *Statistics and computing*, 27(5):1413–1432.

Vehtari, A., Gelman, A., Simpson, D., Carpenter, B., and Bürkner, P.-C. (2021). Rank-normalization, folding, and localization: An improved $\hat{R}$ for assessing convergence of MCMC. *Bayesian analysis*, 1(1):1–28.

Vladimirova, M., Verbeek, J., Mesejo, P., and Arbel, J. (2019). Understanding Priors in Bayesian Neural Networks at the Unit Level. In *International Conference on Machine Learning*, pages 6458–6467.

Wang, C., Tian, M., and Tang, M.-L. (2022). Nonparametric Quantile Regression with Missing Data Using Local Estimating Equations. *Journal of Nonparametric Statistics*, 34(1):164–186.

Wang, W. and Yan, J. (2021). *splines2: Regression Spline Functions and Classes*. R package version 0.4.5.

Watanabe, S. (2013). A widely applicable Bayesian information criterion. *Journal of Machine Learning Research*, 14(Mar):867–897.

Wei, Y. and Yang, Y. (2014). Quantile regression with covariates missing at random. *Statistica Sinica*, pages 1277–1299.

Wickham, H. (2016). *ggplot2: Elegant Graphics for Data Analysis*. Springer-Verlag New York.

Wu, T. Z., Yu, K., and Yu, Y. (2010). Single-Index Quantile Regression. *Journal of Multivariate Analysis*, 101(7):1607–1621.

Xia, Y., Tong, H., Li, W. K., and Zhu, L.-X. (2002). An Adaptive Estimation of Dimension Reduction Space. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 64(3):363–410.

Xie, Y., Cotton, C., and Zhu, Y. (2020). Multiply robust estimation of causal quantile treatment effects. *Statistics in Medicine*, 39(28):4238–4251.

Xu, D., Daniels, M. J., and Winterstein, A. G. (2018). A Bayesian nonparametric approach to causal inference on quantiles. *Biometrics*, 74(3):986–996.

Xu, S. and Majumder, R. (2022). *SPQR: Semi-Parametric Quantile Regression*. R package version 0.1.0.

Xu, S. G. (2021). 2019 U.S. Birth Weight. Open Science Framework. http://doi.org/10.17605/OSF.IO/3GFHE.

Xu, S. G. and Reich, B. J. (2021). Bayesian nonparametric quantile process regression and estimation of marginal quantile effects. *Biometrics*.

Yang, S. and Zhang, Y. (2020). Multiply robust matching estimators of average and quantile treatment effects. *Scandinavian Journal of Statistics*.

Yang, Y. and Tokdar, S. T. (2017). Joint estimation of quantile planes over arbitrary predictor spaces. *Journal of the American Statistical Association*, 112:1107–1120.

Yu, K. and Jones, M. (1998). Local Linear Quantile Regression. *Journal of the American statistical Association*, 93(441):228–237.

Yu, Y. (2021). *siqr: An R Package for Single-Index Quantile Regression*. R package version 0.8.1.

Yu, Y. and Ruppert, D. (2002). Penalized Spline Estimation for Partially Linear Single-Index Models. *Journal of the American Statistical Association*, 97(460):1042–1054.

Yuan, Y., Chen, N., and Zhou, S. (2017). Modeling regression quantile process using monotone B-splines. *Technometrics*, 59:338–350.

Zhang, Z., Chen, Z., Troendle, J. F., and Zhang, J. (2012). Causal inference on quantiles with an obstetric application. *Biometrics*, 68(3):697–706.

Zhou, N., Guo, X., and Zhu, L. (2021). The role of propensity score structure in asymptotic efficiency of estimated conditional quantile treatment effect. *Scandinavian Journal of Statistics*.

Zhou, Y., Wan, A. T. K., and Wang, X. (2008). Estimating Equations Inference with Missing Data. *Journal of the American Statistical Association*, 103(483):1187–1199.

Zhu, L. and Xue, L. (2006). Empirical Likelihood Confidence Regions in a Partially linear single-index model. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 68(3):549–570.

Zigler, C. M., Watts, K., Yeh, R. W., Wang, Y., Coull, B. A., and Dominici, F. (2013). Model feedback in Bayesian propensity score estimation. *Biometrics*, 69(1):263–273.

Zou, Y., Fan, G., and Zhang, R. (2020). Quantile regression and variable selection for partially linear single-index models with missing censoring indicators. *Journal of Statistical Planning and Inference*, 204:80–95.

APPENDICES

APPENDIX

A

SUPPLEMENT TO "BAYESIAN NONPARAMETRIC QUANTILE PROCESS REGRESSION AND ESTIMATION OF MARGINAL QUANTILE EFFECTS"

A.1 Posterior evaluation

The non-parametric model on the conditional PDF of Z given $\boldsymbol{X} = \boldsymbol{x}$ is

$$f_Z(z|\mathbf{X}, \mathscr{W}) = \sum_{k=1}^{K} \theta_k(\mathbf{X}, \mathscr{W}) M_k(z) = \sum_{k=1}^{K} \frac{\exp\{u_k(\mathbf{X}, \mathscr{W})\}}{\sum_{i=1}^{K} \exp\{u_i(\mathbf{X}, \mathscr{W})\}} M_k(z),$$

and the likelihood function is

$$\mathscr{L}(\mathscr{D}|\mathscr{W}) = \prod_{i=1}^{n} f_Z(z_i, \boldsymbol{x}_i|\mathscr{W}) = \prod_{i=1}^{n} \left\{ \sum_{k=1}^{K} \frac{\exp\{u_k(\boldsymbol{x}_i, \mathscr{W})\}}{\sum_{j=1}^{K} \exp\{u_j(\boldsymbol{x}_i, \mathscr{W})\}} M_k(z_i) \right\}.$$

where $\mathscr{D} = \{z_i, \boldsymbol{x}_i\}_{i=1}^{n}$ denotes the observed data. Let $\boldsymbol{\Theta} = \{\mathscr{W}, \sigma_w, \gamma\}$ denote the set of modeling parameters and hyper-parameters, then the posterior of QUINN is

$$f(\boldsymbol{\Theta}|\mathscr{D}) \propto \mathscr{N}^+(\gamma|0, a^2) \prod_{k=1}^{K} \prod_{l=0}^{V} \mathscr{N}(W_{2kl}|0, \gamma^2) \prod_{j=0}^{d} \mathscr{N}^+(\sigma_j|0, a^2)$$
$$\prod_{l=0}^{V} \prod_{j=0}^{d} \mathscr{N}(W_{1lj}|0, \sigma_j^2) \prod_{i=1}^{n} f_Z(z_i, \boldsymbol{x}_i|\mathscr{W})$$

which can be approximated using MCMC methods. Sampling from this posterior is challenging for traditional MCMC methods such as random-walk Metropolis (Metropolis et al. 1953) and Gibbs sampler (Geman and Geman 1984). These methods, although straightforward to implement, do not scale well to complicated posterior with high-dimensional parameter space. The former explores the posterior via inefficient random walks, resulting in low acceptance rate and wasted samples; the latter requires knowing the conditional distribution of each parameter, which can be unrealistic in high-dimensional case.

A.1.1 Hamiltonian Monte Carlo

Hamiltonian Monte Carlo (HMC) (Neal 2011; Betancourt and Girolami 2015; Betancourt 2017) is a variant of MCMC that permits efficient sampling from a high-dimensional target distribution, provided that all model parameters are continuous. It has gained increasing popularity for its recent applications in inference of Bayesian neural networks (Neal 2012). By introducing auxillary variables $\boldsymbol{r}$, HMC transforms the problem of sampling from $f(\boldsymbol{\Theta}|\mathscr{D})$ to sampling from the joint distribution $f(\boldsymbol{r}, \boldsymbol{\Theta}|\mathscr{D}) = f(\boldsymbol{r}|\boldsymbol{\Theta}, \mathscr{D}) f(\boldsymbol{\Theta}|\mathscr{D})$ where $f(\boldsymbol{r}|\boldsymbol{\Theta}, \mathscr{D})$ is the auxiliary distribution often assumed to be multivariate Normal and independent of $\boldsymbol{\Theta}$ and $\mathscr{D}$, i,e. $f(\boldsymbol{r}|\boldsymbol{\Theta}, \mathscr{D}) = f(\boldsymbol{r})$. The joint distribution defines a Hamiltonian

$$H(\boldsymbol{r}, \boldsymbol{\Theta}|\mathscr{D}) = T(\boldsymbol{r}) + V(\boldsymbol{\Theta}|\mathscr{D})$$
$$T(\boldsymbol{r}) \coloneqq -\log f(\boldsymbol{r})$$
$$V(\boldsymbol{\Theta}|\mathscr{D}) \coloneqq -\log f(\boldsymbol{\Theta}|\mathscr{D}).$$

which can be used to generate states, i.e. samples of $\boldsymbol{\Theta}$ and $\boldsymbol{r}$, by simulating the Hamiltonian dynamics

$$\frac{\partial \boldsymbol{\Theta}}{\partial t} = \frac{\partial T}{\partial \boldsymbol{r}}, \ \frac{\partial \boldsymbol{r}}{\partial t} = \frac{\partial V}{\partial \boldsymbol{\Theta}}.$$

At any state $(\boldsymbol{\Theta}_t, \boldsymbol{r}_t)$, HMC proposes the next state $(\boldsymbol{\Theta}_{t+L\Delta t}, \boldsymbol{r}_{t+L\Delta t})$ by simulating Hamiltonian dynamics for time $L\Delta t$, which is approximated by applying the leapfrog algorithm L times each with step size Δt. Starting from an initial state, this process is repeated and the visited states form a Markov chain. Compared to random-walk Metropolis, HMC explores the target distribution more efficiently by using gradient of the log-posterior to direct each transition of the Markov chain. Although each step is more computationally expensive than a Metropolis proposal, the Markov chain produced by HMC often yields more distant samples and significantly higher acceptance rate. Although HMC has a high potential, its practical performance depends highly on the values of L and Δt. Poor choice of either parameter will result in unsatisfactory exploration of the posterior. In this paper, instead of using the original HMC which only allows manual setting of L and Δt, we use the No-U-Turn Sampler (NUTS). NUTS is an extension to HMC that implements automatic tuning of L. Furthermore, we use the dual averaging algorithm to adaptively select Δt. A detailed description of NUTS with dual averaging is presented in Algorithm 6 of Hoffman and Gelman (2014).

NUTS is implemented in many probabilistic programming framework, such as PyMC3 (Salvatier et al. 2016) and Stan (Stan Development Team 2019). Computational complexity of an HMC implementation is contingent on gradient calculation of the log-posterior, which the aforementioned high-level frameworks handle via automatic differentiation. In our experiment, we observe that automatic differentiation can be extremely time consuming for a posterior as high-dimensional as ours. As a result, we use a low-level R implementation (Monnahan and Kristensen 2018) that accepts analytic gradients which we manually calculate.

A.1.2 Reparametrization and transformation

The hierarchical Gaussian priors $W_{1vw} \overset{indep}{\sim} \mathcal{N}(0, \sigma_w^2)$, $\sigma_w \overset{iid}{\sim} \mathcal{N}^+(0, a^2)$ introduce strong correlation between W_{1vw} and σ_w in the posterior, especially when the data size is small.

To alleviate this issue, we consider a reparametrization:

$$B_{1vw} \overset{iid}{\sim} \mathcal{N}(0,1),\ \sigma_w \overset{iid}{\sim} \mathcal{N}^+(0,a^2),\ W_{1vw} = \sigma_w B_{1vw}$$

where B_{1vw} can be considered as standardized weights. Because B_{1vw} and σ_w follow independent prior distributions, they are marginally uncorrelated in the posterior. Their coupling is instead introduced in the likelihood function. Such a parameterization is called non-centered. Non-centered parameterization leads to simpler posterior geometries, thus increasing the efficiency of HMC. Similarly, $W_{2vw} \overset{indep}{\sim} \mathcal{N}(0,\gamma^2)$, $\gamma \sim \mathcal{N}^+(0,a^2)$ can be reparameterized as

$$B_{2vw} \overset{iid}{\sim} \mathcal{N}(0,1),\ \gamma \sim \mathcal{N}^+(0,a^2),\ W_{2vw} = \gamma B_{2vw}.$$

HMC requires $\boldsymbol{\Theta}$ to lie in an unconstrained space, and thus every parameter that has a natural constraint needs to be transformed to an unconstrained variable. After unconstrained posterior samples are drawn, they can be back-transformed to the constrained space. In $\boldsymbol{\Theta}$, the scale parameters σ_v and γ are naturally constrained to be positive. Therefore we work with their log-transformations $\tilde{\sigma}_v = \log\sigma_v$ and $\tilde{\gamma} = \log\gamma$ with transformed prior distributions

$$f(\tilde{\sigma}_v) = \mathcal{N}^+(\exp(\tilde{\sigma}_v)|0,a^2)\exp(\tilde{\sigma}_v) \quad \text{and} \quad f(\tilde{\gamma}) = \mathcal{N}^+(\exp(\tilde{\gamma})|0,a^2)\exp(\tilde{\gamma}).$$

Let $\mathscr{B} = \{\beta_{uvw}\}$ and $\tilde{\boldsymbol{\Theta}} = \{\mathscr{B}, \tilde{\sigma}_w, \tilde{\gamma}\}$, then the posterior after non-centered reparameterization and constraint transformation is

$$\begin{aligned} f(\tilde{\boldsymbol{\Theta}}|\mathscr{D}) \propto \mathcal{N}^+(\exp(\tilde{\gamma})|0,a^2)\exp(\tilde{\gamma}) \prod_{j=1}^{J}\prod_{l=0}^{V}\mathcal{N}(B_{2ml}|0,1)\prod_{j=0}^{d}\mathcal{N}^+(\exp(\tilde{\sigma}_j)|0,a^2)\exp(\tilde{\sigma}_j) \\ \prod_{l=0}^{V}\prod_{j=0}^{d}\mathcal{N}(B_{1lj}|0,1)\prod_{i=1}^{n} f_Z(z_i,\boldsymbol{x}_i|\mathscr{W}), \end{aligned}$$

where $W_{1vw} = \exp(\tilde{\sigma}_w)B_{1vw}$ and $W_{2vw} = \exp(\tilde{\gamma})B_{2vw}$ in the likelihood function.

A.1.3 Analytic gradient

In this section, we provide analytic formulas for computing the gradient of $\log f(\tilde{\boldsymbol{\Theta}}|\mathscr{D})$, which NUTS uses to generate samples of $\tilde{\boldsymbol{\Theta}}$. To start with, let $\mathbf{X}$ denote the observed covariate matrix, $\mathbf{M}(\boldsymbol{z})$ denote the M-spline matrix of transformed response vector $\boldsymbol{z}$, $\mathbf{1}$ denote a

column vector of 1's, $\boldsymbol{\sigma}$ denote the vector with elements σ_w, and $\mathbf{B}_u$ denote the matrix with elements B_{uvw}. The log-likelihood function parametrized by $\tilde{\boldsymbol{\Theta}}$ can be written in a compact form using matrix notation

$$\ell(\mathscr{D}|\tilde{\boldsymbol{\Theta}}) = \sum_{i=1}^{n}\left(\log\left[\sum_{j=1}^{J}\exp\left\{u_j(\boldsymbol{x}_i,\tilde{\boldsymbol{\Theta}})\right\}M_{m,r}(z_i)\right] - \log\left[\sum_{j=1}^{J}\exp\left\{u_j(\boldsymbol{X}_i,\tilde{\boldsymbol{\Theta}})\right\}\right]\right)$$

$$= \mathbf{1}^T\left(\log\left[\exp\left\{\mathbf{U}(\mathbf{X},\tilde{\boldsymbol{\Theta}})\right\}\odot\mathbf{M}(z)\mathbf{1}\right] - \log\left[\exp\left\{\mathbf{U}(\mathbf{X},\tilde{\boldsymbol{\Theta}})\right\}\mathbf{1}\right]\right)$$

where

$$\mathbf{U}(\mathbf{X},\tilde{\boldsymbol{\Theta}}) = \left(\mathbf{1} \;\middle|\; \phi\left\{\tilde{\mathbf{X}}\tilde{\mathbf{D}}\mathbf{B}_1\right\}\right)\left[\exp(\tilde{\gamma})\mathbf{B}_2\right],\ \tilde{\mathbf{X}} = \left(\mathbf{1} \;\middle|\; \mathbf{X}\right),$$

and $\tilde{\mathbf{D}}$ is the diagonal matrix with diagonal entries $\exp(\tilde{\boldsymbol{\sigma}})$. The log-prior can be written as

$$f(\tilde{\boldsymbol{\Theta}}) \propto -\frac{\exp(2\tilde{\gamma})a^2}{\pi} + \tilde{\gamma} - \frac{\mathrm{vec}(\mathbf{B}_2)^T\mathrm{vec}(\mathbf{B}_2)}{2} - \mathbf{1}^T\left[\frac{\exp(2\tilde{\boldsymbol{\sigma}})a^2}{\pi} - \tilde{\boldsymbol{\sigma}}\right] - \frac{\mathrm{vec}(\mathbf{B}_1)^T\mathrm{vec}(\mathbf{B}_1)}{2}.$$

where vec(·) denotes the vectorization operator. Finally, the gradient formula of the log-posteior with respect to each parameter, expressed using matrix notation, is given by

$$\begin{aligned}
\frac{\partial\log f(\tilde{\boldsymbol{\Theta}}|\mathscr{D})}{\partial\mathbf{B}_1} &= \exp(\tilde{\gamma})\tilde{\mathbf{D}}\tilde{\mathbf{X}}^T(\mathbf{V}_1\mathbf{V}_3 - \mathbf{V}_2\mathbf{V}_3) - \mathbf{B}_1\\
\frac{\partial\log f(\tilde{\boldsymbol{\Theta}}|\mathscr{D})}{\partial\mathbf{B}_2} &= \exp(\tilde{\gamma})\mathbf{V}_0^T\left(\mathbf{V}_1\left[\exp\left\{\mathbf{U}(\mathbf{X},\tilde{\boldsymbol{\Theta}})\right\}\odot\mathbf{M}(z)\right] - \mathbf{V}_2\exp\left\{\mathbf{U}(\mathbf{X},\tilde{\boldsymbol{\Theta}})\right\}\right) - \mathbf{B}_2\\
\frac{\partial\log f(\tilde{\boldsymbol{\Theta}}|\mathscr{D})}{\partial\tilde{\boldsymbol{\sigma}}} &= \exp(\tilde{\gamma})\left[\mathrm{diag}\left\{\tilde{\mathbf{X}}^T(\mathbf{V}_1\mathbf{V}_3 - \mathbf{V}_2\mathbf{V}_4)\mathbf{B}_1^T\right\}\right]\odot\exp(\tilde{\boldsymbol{\sigma}}) - \frac{2a^2}{\pi}\exp(2\tilde{\boldsymbol{\sigma}}) + \mathbf{1}\\
\frac{\partial\log f(\tilde{\boldsymbol{\Theta}}|\mathscr{D})}{\partial\tilde{\gamma}} &= \exp(\tilde{\gamma})\Bigg(\mathrm{tr}\left\{\mathbf{V}_0\mathbf{B}_2\left[\exp\left\{\mathbf{U}(\mathbf{X},\tilde{\boldsymbol{\Theta}})\right\}\odot\mathbf{M}(z)\right]^T\mathbf{V}_1\right\}\\
&\qquad - \mathrm{tr}\left\{\mathbf{V}_0\mathbf{B}_2\exp\left\{\mathbf{U}(\mathbf{X},\tilde{\boldsymbol{\Theta}})\right\}^T\mathbf{V}_2\right\}\Bigg) - \frac{2a^2}{\pi}\exp(2\tilde{\gamma}) + 1
\end{aligned}$$

where

$$\begin{aligned}
\mathbf{V}_0 &= \left(\mathbf{1} \;\middle|\; \phi\left\{\tilde{\mathbf{X}}\tilde{\mathbf{D}}\mathbf{B}_1\right\}\right) \\
\mathbf{V}_1 &= \text{diag}\left\{\mathbf{1} \oslash \left[\exp\left\{\mathbf{U}(\mathbf{X}, \tilde{\boldsymbol{\Theta}})\right\} \odot \mathbf{M}(z)\mathbf{1}\right]\right\} \\
\mathbf{V}_2 &= \text{diag}\left\{\mathbf{1} \oslash \left[\exp\left\{\mathbf{U}(\mathbf{X}, \tilde{\boldsymbol{\Theta}})\right\} \mathbf{1}\right]\right\} \\
\mathbf{V}_3 &= \left[\exp\left\{\mathbf{U}(\mathbf{X}, \tilde{\boldsymbol{\Theta}})\right\} \odot \mathbf{M}(z)\bar{\mathbf{B}}_2^T\right] \odot \phi'\left(\tilde{\mathbf{X}}\tilde{\mathbf{D}}\mathbf{B}_1\right) \\
\mathbf{V}_4 &= \left[\exp\left\{\mathbf{U}(\mathbf{X}, \tilde{\boldsymbol{\Theta}})\right\} \bar{\mathbf{B}}_2^T\right] \odot \phi'\left(\tilde{\mathbf{X}}\tilde{\mathbf{D}}\mathbf{B}_1\right),
\end{aligned}$$

$\odot$ denotes element-wise multiplication, $\oslash$ denotes element-wise division, and $\bar{\mathbf{B}}_u$ is $\mathbf{B}_u$ after removing the first row.

A.1.4 Model estimation

It is well-known that FNN suffers from over-parameterization, which makes the weight parameters highly non-identifiable. In practice, MCMC for individual weights might not even converge, making Bayesian inference of the weight parameters impossible. Let $\mathscr{W}^{(t)}$, $t = 1, ..., T$ denote the t-th posterior sample of $\mathscr{W}$. In this study, instead of using the posterior estimates (e.g. posterior means) of the weight parameters to calculate a single estimate of $F_Z(z|\mathbf{X}, \hat{\mathscr{W}})$

$$\hat{\mathscr{W}} = \frac{1}{T}\sum_{t=1}^{T} \mathscr{W}^{(t)}$$

we estimate $F_Z(z|\boldsymbol{x})$ using its posterior mean

$$\hat{F}_Z(z|\boldsymbol{x}) = \frac{1}{T}\sum_{t=1}^{T} F(z|\mathbf{X}, \mathscr{W}^{(t)}).$$

Convergence of MCMC can be checked using the trace plot of $F_Z(z|\mathbf{X}, \mathscr{W}^{(t)})$ for some $(z, \boldsymbol{x})$.

A.1.5 Convergence diagnostics

To monitor the convergence of NUTS, we simulate multiple independent chains and inspect the trace plots of data log-likelihood. We choose to monitor the log-likelihood because it is efficient to calculate and represents the goodness-of-fit of our non-parametric model. We do not monitor individual weight parameters because they suffer from non-identifiability due to the over-parametrization of FNN; it is likely that their trace plots display multimodality

rather than converging to a single distribution. As an illustration, Fig. A.1 plots the traces of log-likelihood of four chains for one replicate of Simulation 1–4, after discarding burn-ins. The plot shows a good mixing of the four chains, all converging to the same target distribution.

In addition to trace plots, we also utilize various diagnostic statistics to more precisely assess convergence. Vehtari et al. (2021) recently propose to inspect the values of bulk and tail effective sample size (ESS) together with an improved Gelman-Rubin $\hat{R}$ to assess MCMC convergence. Following their recommendation, we consider the chains converge only if both bulk and tail ESS are greater than 100, and $\hat{R}$ is less than 1.05. Calculation of these statistics are carried out by functions implemented in the R package **rstan**.

A.2 Accumulative local effects

In this section, we first briefly summarize how the main and interaction ALE can be estimated using sample data; the full description can be seen in Apley and Zhu (2020). We then explain how VI scores can be estimate based on the ALE estimates.

Let $x_{i,j}$ and $\boldsymbol{x}_{i,\setminus j}$ denote the ith observation of jth covariate and all other covariates respectively. The sample range of X_j is partitioned into K intervals $\{N_j(k) = (z_{k-1,j}, z_{k,j}] : k = 1,2,...,K\}$ where $z_{k,j}$ are chosen as the k/K-th sample percentile if X_j is continuous and the unique values otherwise. Then the uncentered effect $\bar{Q}_j^U(\tau, x_j)$ can be estimated by

$$\hat{\bar{Q}}_j^U(\tau, x_j) = \sum_{k=1}^{k_j(x_j)} \frac{1}{n_j(k)} \sum_{\{i: x_{i,j} \in N_j(k)\}} \big[q_j(\tau, z_{k,j}, \boldsymbol{x}_{i,\setminus j}) - q_j(\tau, z_{k-1,j}, \boldsymbol{x}_{i,\setminus j})\big],$$

where $k_j(x_j)$ index the interval into which x_j falls, and $n_j(k)$ denotes the number of sample observations $N_j(k)$ contains such that $n = \sum_{k=1}^{k} n_j(k)$. Finally, $\bar{Q}_j(\tau, x_j)$ can be estimated by mean-centering $\hat{\bar{Q}}_j^U(\tau, x_j)$, i.e.

$$\hat{\bar{Q}}_j(\tau, x_j) = \hat{\bar{Q}}_j^U(\tau, x_j) - \frac{1}{n} \sum_{k=1}^{K} n_j(k) \bar{Q}_j^U(\tau, z_{k,j}).$$

For any pair of covariates $\{X_j, X_l\}$, let $\boldsymbol{x}_{i,\{j,l\}}$ denote ith observation vector of jth and lth covariate, and $\boldsymbol{x}_{i,\setminus\{j,l\}}$ denote all other covariates. The Cartesian product of sample ranges of X_j and X_l can be partitioned into K^2 rectangular cells $N_{\{j,l\}}(k,m) = (z_{k-1,j}, z_{k,j}] \times (z_{l-1,j}, z_{l,j}]$.

Then the uncentered effect $\bar{Q}^U_{j,l}(\tau, x_j, x_l)$ can be estimated by

$$\begin{aligned}\hat{\bar{Q}}^U_{jl}(\tau, x_j, x_l) = \sum_{k=1}^{k_j(x_j)} \sum_{m=1}^{k_l(x_l)} \frac{1}{n_{\{j,l\}}(k,m)} \sum_{\{i:\boldsymbol{x}_{i,\{j,l\}} \in N_{\{j,l\}}(k,m)\}} \Big[& q_{jl}(\tau, z_{k,j}, z_{m,l}, \boldsymbol{x}_{i,\setminus\{j,l\}}) \\ & - q_{jl}(\tau, z_{k-1,j}, z_{m,l}, \boldsymbol{x}_{i,\setminus\{j,l\}}) - \big\{ q_{jl}(\tau, z_{k,j}, z_{m-1,l}, \boldsymbol{x}_{i,\setminus\{j,l\}}) \\ & - q_{jl}(\tau, z_{k-1,j}, z_{m-1,l}, \boldsymbol{x}_{i,\setminus\{j,l\}}) \big\} \Big],\end{aligned}$$

where $k_j(x_j), k_l(x_l)$ index the cell into which (x_j, x_l) falls, and $n_{\{j,l\}}(k,m)$ denotes the number of sample observations $N_{\{j,l\}}(k,m)$ contains such that $n = \sum_{k=1}^{k} \sum_{m=1}^{K} n_{\{j,l\}}(k,m)$. Similarly, $\bar{Q}_{jl}(\tau, x_j, x_l)$ can be estimated by mean-centering $\hat{\bar{Q}}^U_{j,l}(\tau, x_j, x_l)$, i.e.,

$$\hat{\bar{Q}}_{jl}(\tau, x_j, x_l) = \hat{\bar{Q}}^U_{jl}(\tau, x_j, x_l) - \frac{1}{n} \sum_{k=1}^{K} \sum_{m=1}^{K} n_{\{j,l\}}(k,m) \hat{\bar{Q}}^U_{jl}(\tau, z_{k,j}, z_{m,l}).$$

Finally, interaction ALE is estimated by $\hat{\bar{Q}}^I_{j,l}(\tau, x_j, x_l) = \hat{\bar{Q}}_{j,l}(\tau, x_j, x_l) - \hat{\bar{Q}}_j(\tau, x_l) - \hat{\bar{Q}}_j(\tau, x_l)$.

Following its definition in Section 3, $\mathrm{VI}_j(\tau)$ can be estimated by the sample standard deviation or the sample range of $\hat{\bar{Q}}_j(\tau, z_{k,j})$, i.e.,

$$\widehat{\mathrm{VI}}_j(\tau) = \begin{cases} \sqrt{\frac{1}{K} \sum_{k=1}^{K} \Big[\hat{\bar{Q}}_j(\tau, z_{k,j}) - \frac{1}{K} \sum_{k=1}^{K} \hat{\bar{Q}}_j(\tau, z_{k,j}) \Big]^2} & \text{if } X_j \text{ is continuous} \\ \Big\{ \max_k \Big[\hat{\bar{Q}}_j(\tau, z_{k,j}) \Big] - \min_k \Big[\hat{\bar{Q}}_j(\tau, z_{k,j}) \Big] \Big\} / 4 & \text{if } X_j \text{ is categorical} \end{cases}.$$

By analogy, $\mathrm{VI}_{jl}(\tau)$ can be estimated by the sample standard deviation or the sample range of $\hat{\bar{Q}}^I_{jl}(\tau, z_{k,j}, z_{m,l})$.

A.3 Implementation detail

In this section, we provide implementation details of the simulation study for the competing methods. For MCQRNN, we consider neural networks with a single hidden layer, $V \in \{3, 5, 8, 10, 15\}$ hidden neurons, and weight penalty coefficients $\lambda \in \{e^{-2}, e^{-3}, ..., e^{-6}\}$. We select the best configuration of V and λ using 5-fold cross-validation. For NPSQR and NPDFSQR, we follow the guidelines provided by Das and Ghosal (2018) and first transform the response variable and covariate(s) into unit intervals using min-max normalization. The response variable and covariate(s) are then expanded using quadratic B-splines with same

number of equidistant knots, denoted as p_{DG}. We fit NPSQR with $p_{\text{DG}} \in \{3,4,...,10\}$ and NPDFSQR with $p_{\text{DG}} \in \{5,6,...,10\}$. The optimal p_{DG} for either model is chosen based on the AIC in which maximum likelihood estimates are replaced by posterior means. SeriesCDE contains four tuning parameters: the number of components of the series expansion in the Y direction N_Y, the number of components of the series expansion in the $\boldsymbol{X}$ direction $N_{\boldsymbol{X}}$, the bandwidth parameter ϵ of the Gaussian kernel for constructing the Gram matrix of $\boldsymbol{X}$, and the smoothness parameter δ that controls the bumpiness of the estimated conditional density function. The tuning grids are set as $N_{\boldsymbol{X}}, N_Y \in \{1,2,...,n\}$, $\epsilon \in \{e^{-7}, e^{-6.5}, ..., e^{3}\}$, and $\delta \in \{0, 0.05, ..., 0.5\}$. Following the guidelines provided by Izbicki and Lee (2016), we first select the best configuration of $N_{\boldsymbol{X}}$, N_Y, and ϵ using 5-fold cross-validation. We then tune δ using again 5-fold cross-validation while fixing $N_{\boldsymbol{X}}$, N_Y, and ϵ at their optimal values.

Since NPDFSQR, and seriesCDE model the distribution function, their results have to be converted to estimates of the quantile function. For NPDFSQR, once we obtain the non-parametric CDF estimate of the transformed response $\hat{F}_Z(z|\boldsymbol{x})$, we evaluate it at 101 equidistant grid-points on the unit interval for each $\boldsymbol{x}$. The conditional quantile function $Q_Z(\tau|\boldsymbol{x})$, $\tau \in (0,1)$ can then be estimated by interpolation using $\left\{\hat{F}_Z(z_i, \boldsymbol{x})\right\}_{i=1}^{101}$ as input values and the aforementioned 101 equidistant grid-points as functional output values. Finally, the quantile function of the original response $Q_Y(\tau|\boldsymbol{x})$ can be estimated by reverting the min-max normalization. For seriesCDE, we first convert the non-parametric density function estimate $\hat{f}_Y(y|\boldsymbol{x})$ to $\hat{F}_Y(y|\boldsymbol{x})$ using numerical integration (e.g. trapezoidal rule). We then estimate $Q_Y(\tau|\boldsymbol{x})$ using the aforementioned interpolation approach.

For NPSQR, NPDFSQR, and seriesCDE, the above parameters are used for Simulation 1–4. However, for MCQRNN, we have slightly different tuning parameters for Simulation 4; we consider neural networks with either a single or two hidden layers, $V \in \{3,5,8,10,15\}$ hidden neurons for each hidden layer, and weight penalty coefficients $\lambda \in \{e^{-2}, e^{-3}, ..., e^{-7}\}$. The best configuration of number of layers, V, and λ are selected using 5-fold cross-validation. We do not consider neural networks with more than two hidden layers as they are not currently implemented in **qrnn**.

A.4 Additional results

In this section, we present additional results from the simulation study. Table A.1 gives the average RMISE for all methods, sample sizes and simulation designs. Fig. A.1 plots

Table A.1: Simulation results: Average RMISE_{QP} over 100 replicates with standard error in parentheses, and the smallest error in each row is in bold.

Design	n	QUINN	MCQRNN	NPSQR	NPDFSQR	seriesCDE
	50	**0.86** (0.22)	1.11 (1.98)	1.13 (0.26)	1.15 (0.26)	1.11 (0.29)
1	100	**0.62** (0.11)	0.65 (0.15)	1.00 (0.19)	0.89 (0.17)	0.87 (0.19)
	200	0.50 (0.09)	**0.47** (0.11)	0.96 (0.18)	0.74 (0.16)	0.71 (0.19)
	50	**0.80** (0.16)	1.18 (2.28)	1.18 (0.28)	1.20 (0.31)	0.90 (0.20)
2	100	**0.60** (0.10)	0.72 (0.13)	1.19 (0.22)	0.93 (0.19)	0.74 (0.19)
	200	**0.48** (0.06)	0.53 (0.11)	1.03 (0.18)	0.76 (0.15)	0.57 (0.15)
	50	**1.19** (0.32)	1.39 (0.81)	2.23 (1.78)	2.68 (2.25)	1.23 (0.53)
3	100	**0.93** (0.21)	**0.94** (0.21)	2.33 (1.83)	4.04 (2.62)	0.96 (0.27)
	200	0.82 (0.27)	**0.71** (0.34)	2.16 (1.20)	3.66 (2.29)	0.86 (0.27)
4	200	**2.47** (0.25)	3.21 (1.58)	-	-	3.37 (0.21)

the posterior samples of the log-likelihood for several independent MCMC chains for one dataset generated from each simulation design. Figure A.2 plots the relationship between out-of-sample prediction error and WAIC for QUINN for each simulation design. For each simulated dataset, we fit the model for nine combinations of the tuning parameters (p and V) and record the WAIC; boxplots of RMISE of 100 simulated datasets for each combination of the tuning parameters, grouped by the rank of their WAIC, are shown in Figure A.2. Figures A.3 and A.4 plot the fitted quantile curves and density functions for one dataset from the first two simulation designs. The remaining figures plot interaction surfaces for eight simulated datasets from the fourth simulation scenario.

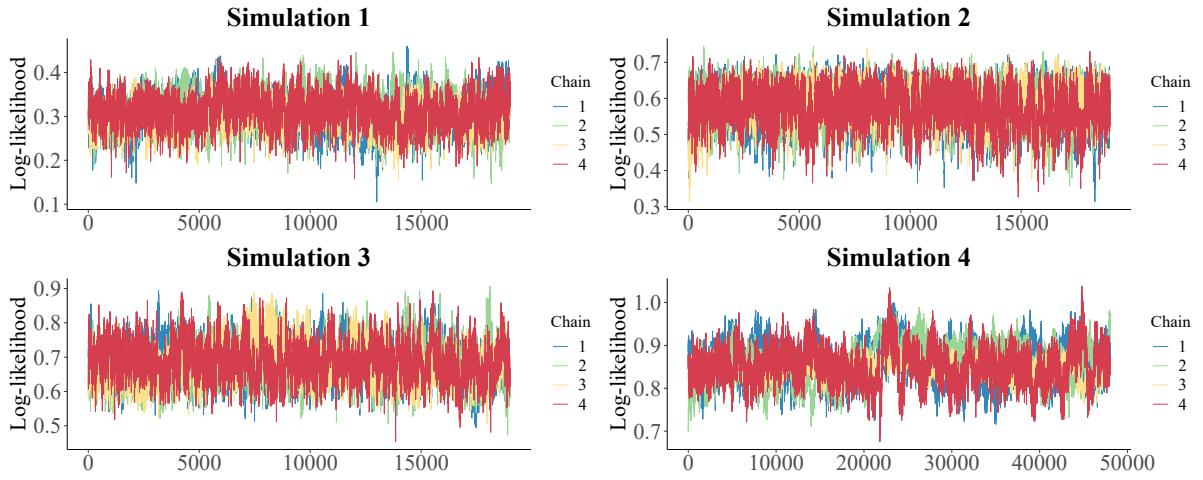


Figure A.1: Trace plot of log-likelihood showing convergence and good mixing of MCMC chains.

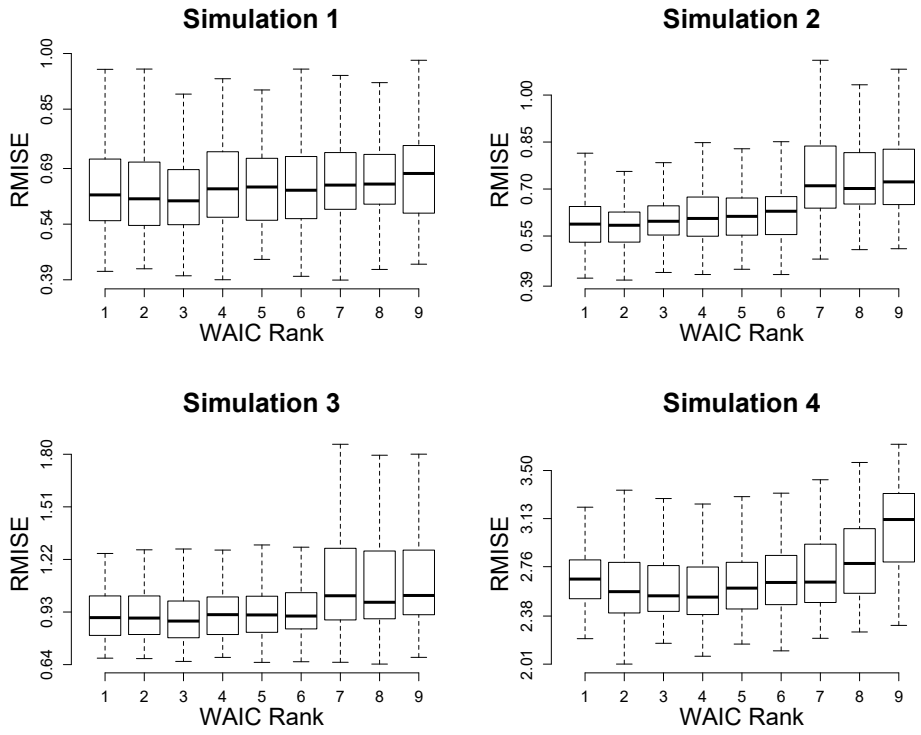


Figure A.2: Distribution of RMISE conditioned on rank of WAIC, constructed using 100 replicates of each simulation study.

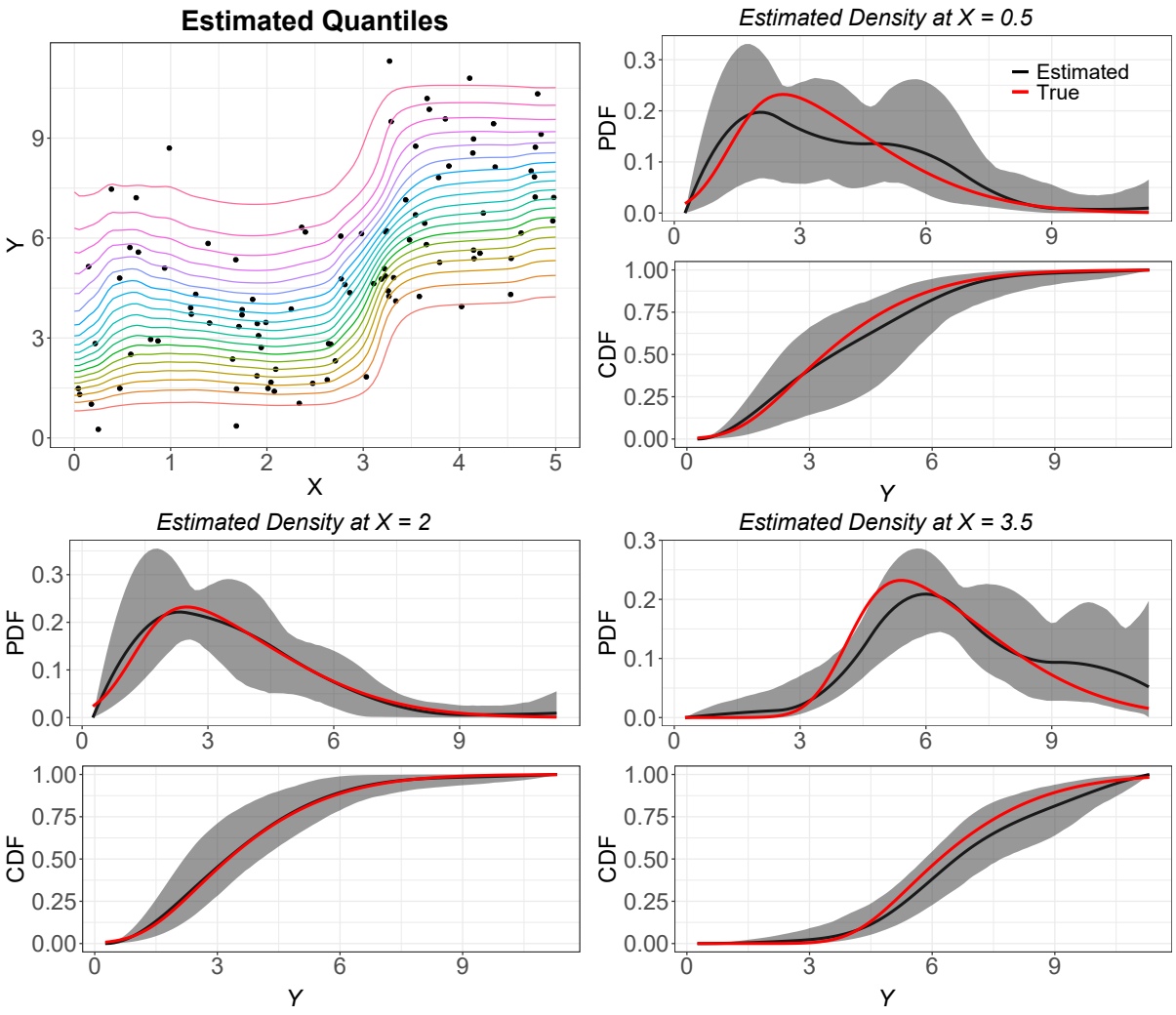


Figure A.3: Posterior estimates of quantile curves and conditional density for Simulation 1. Gray shade represents 95% credible bands.

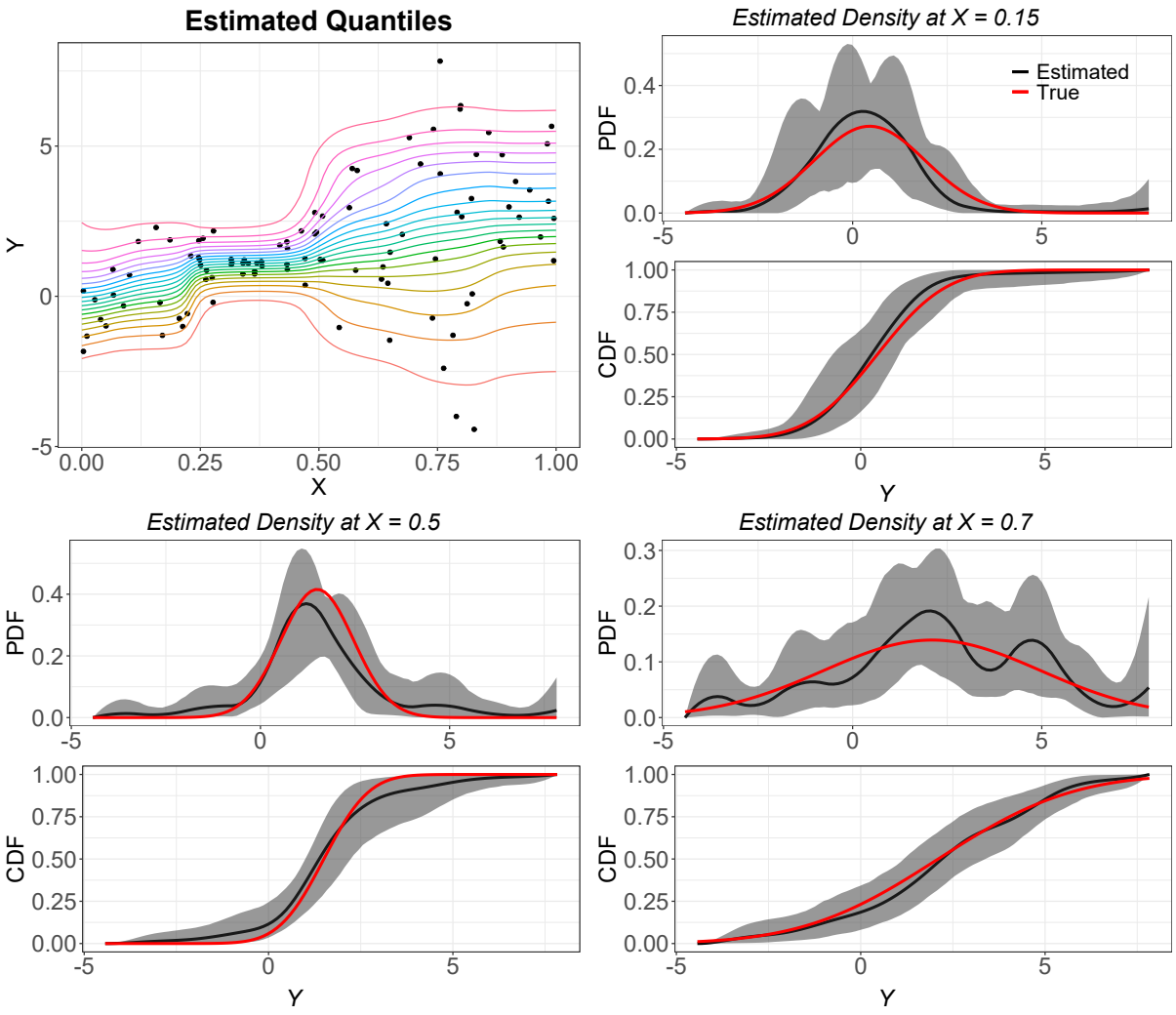


Figure A.4: Posterior estimates of quantile curves and conditional density for Simulation 2. Gray shade represents 95% credible bands.

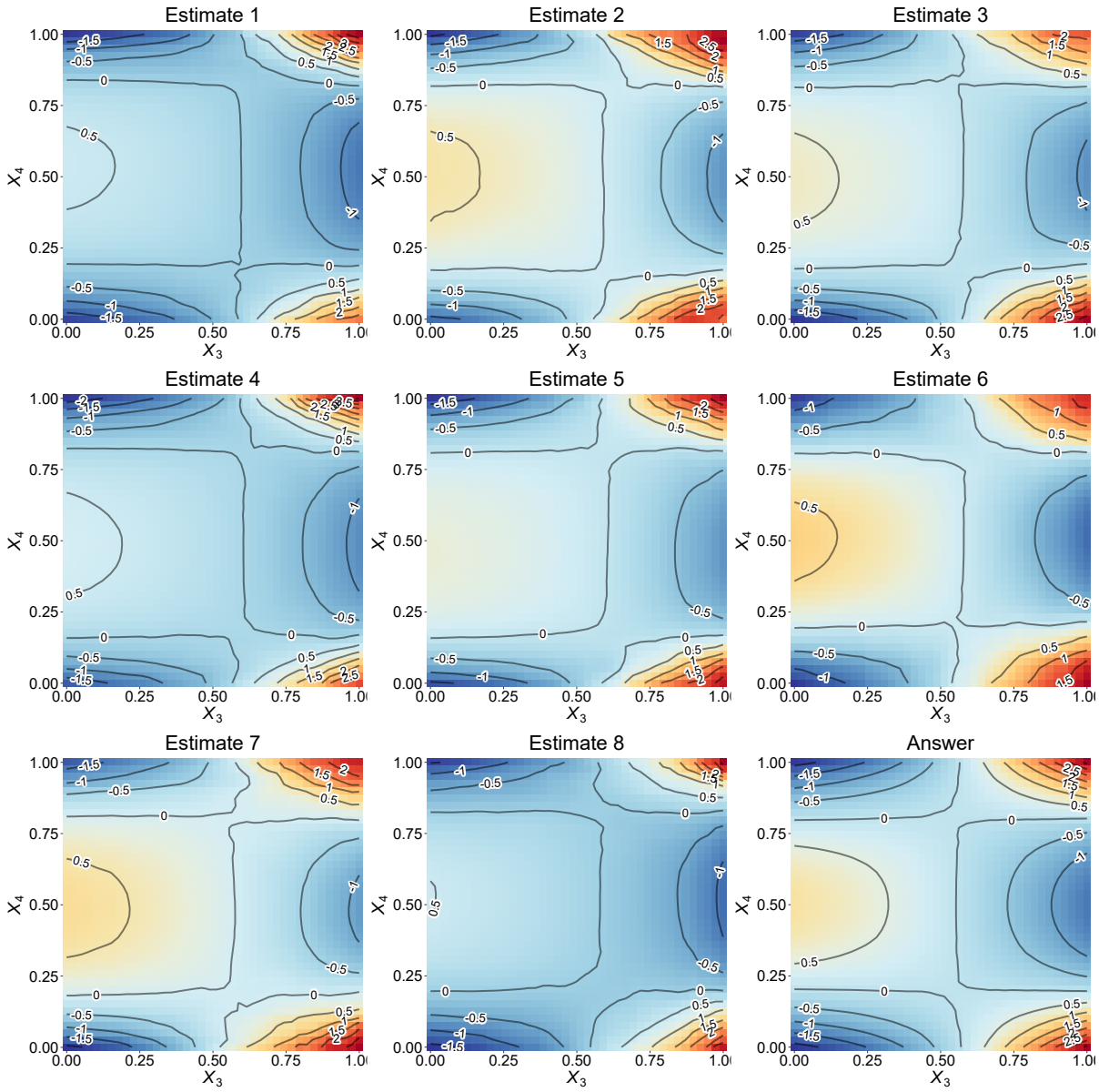


Figure A.5: Marginal interaction effect between X_3 and X_4. Estimate 1–8 are posterior mean ALE interaction effect $\hat{\tilde{Q}}^I_{34}(\tau, x_3, x_4)$ of 8 replicates at quantile level $\tau = 0.05$. "Answer" represents the ground truth.

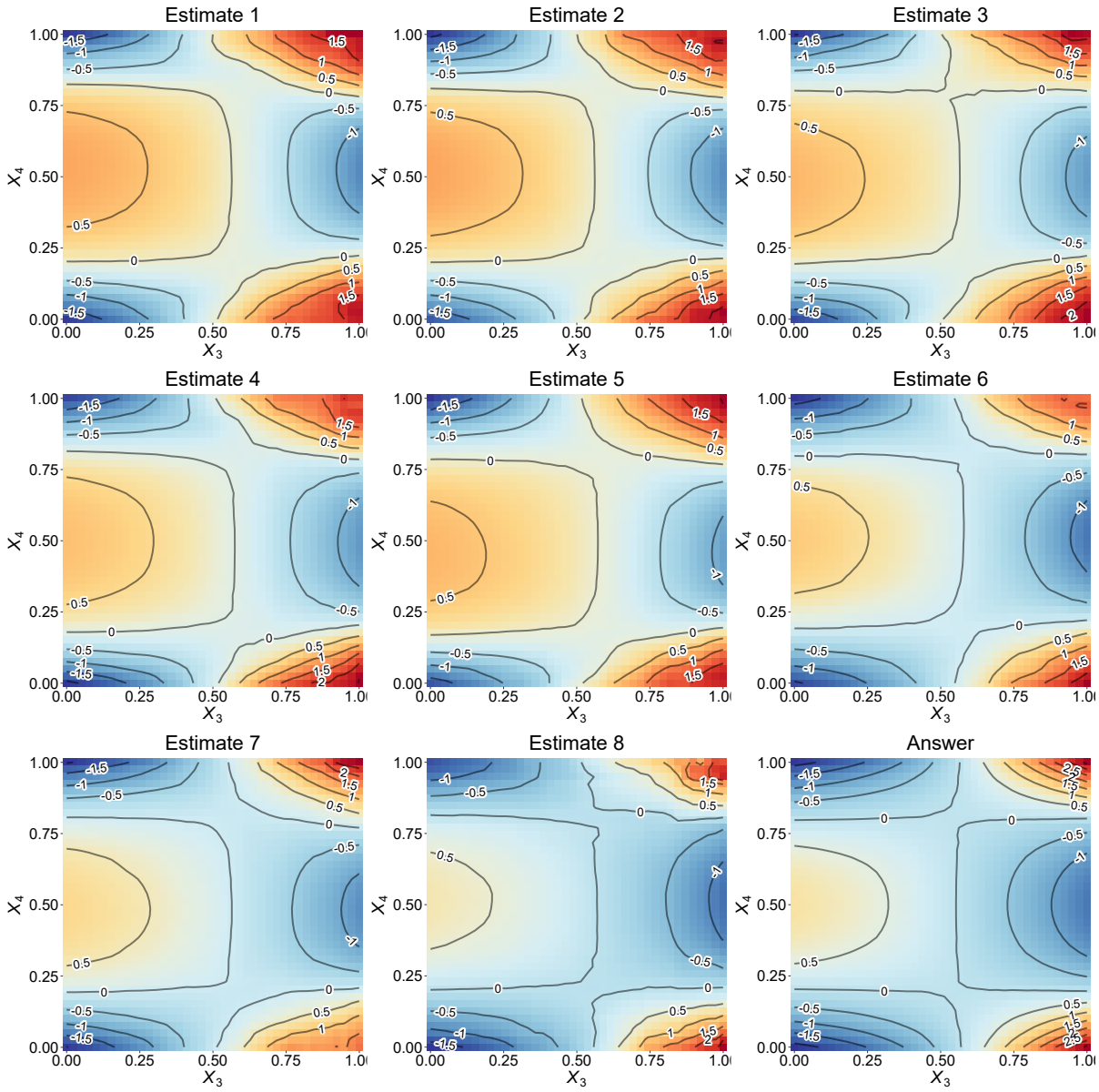


Figure A.6: Marginal interaction effect between X_3 and X_4. Estimate 1–8 are posterior mean ALE interaction effect $\hat{\tilde{Q}}^I_{34}(\tau, x_3, x_4)$ of 8 replicates at quantile level $\tau = 0.5$. “Answer" represents the ground truth.

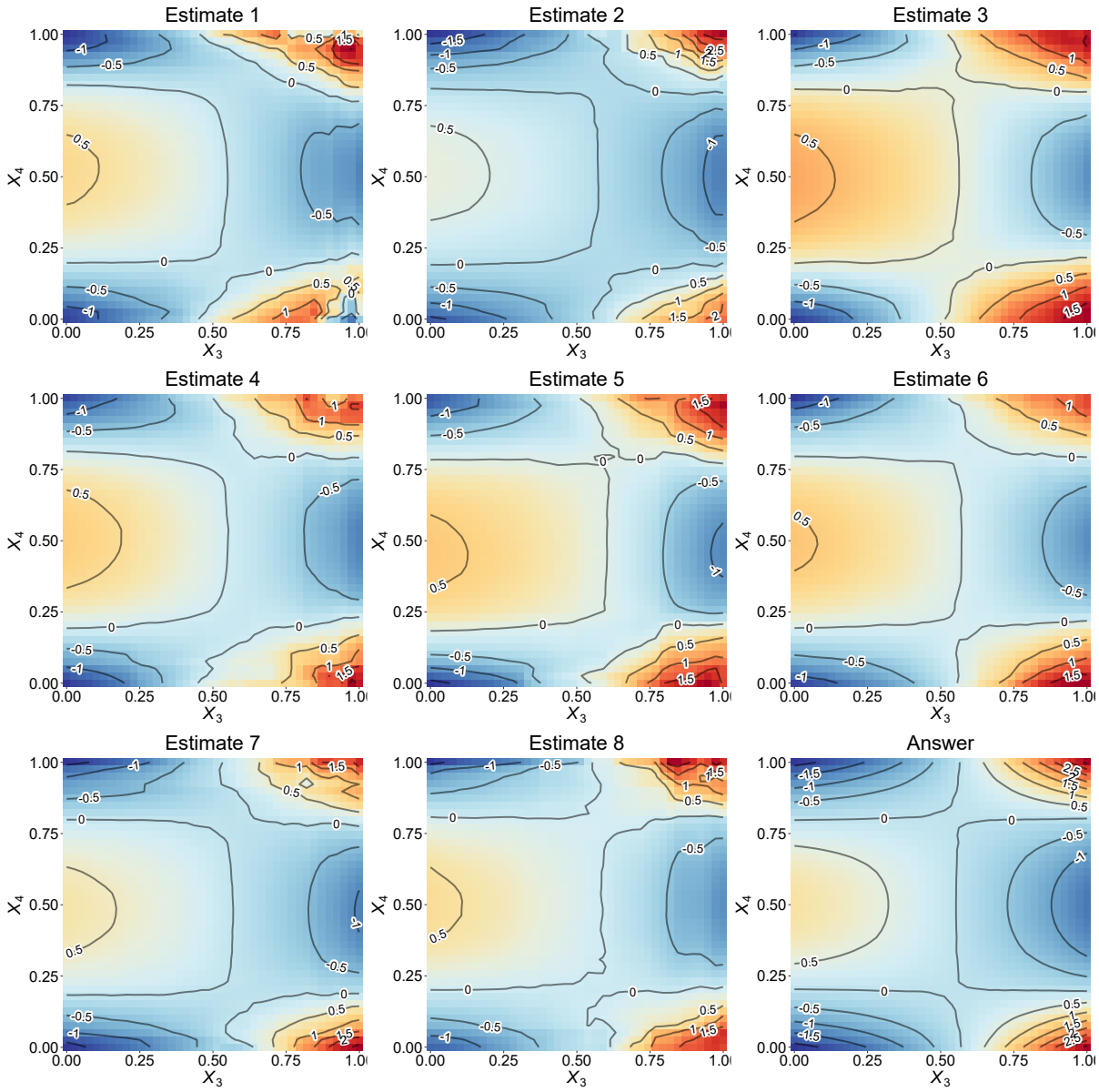


Figure A.7: Marginal interaction effect between X_3 and X_4. Estimate 1–8 are posterior mean ALE interaction effect $\hat{\bar{Q}}^I_{34}(\tau, x_3, x_4)$ of 8 replicates at quantile level $\tau = 0.95$. "Answer" represents the ground truth.

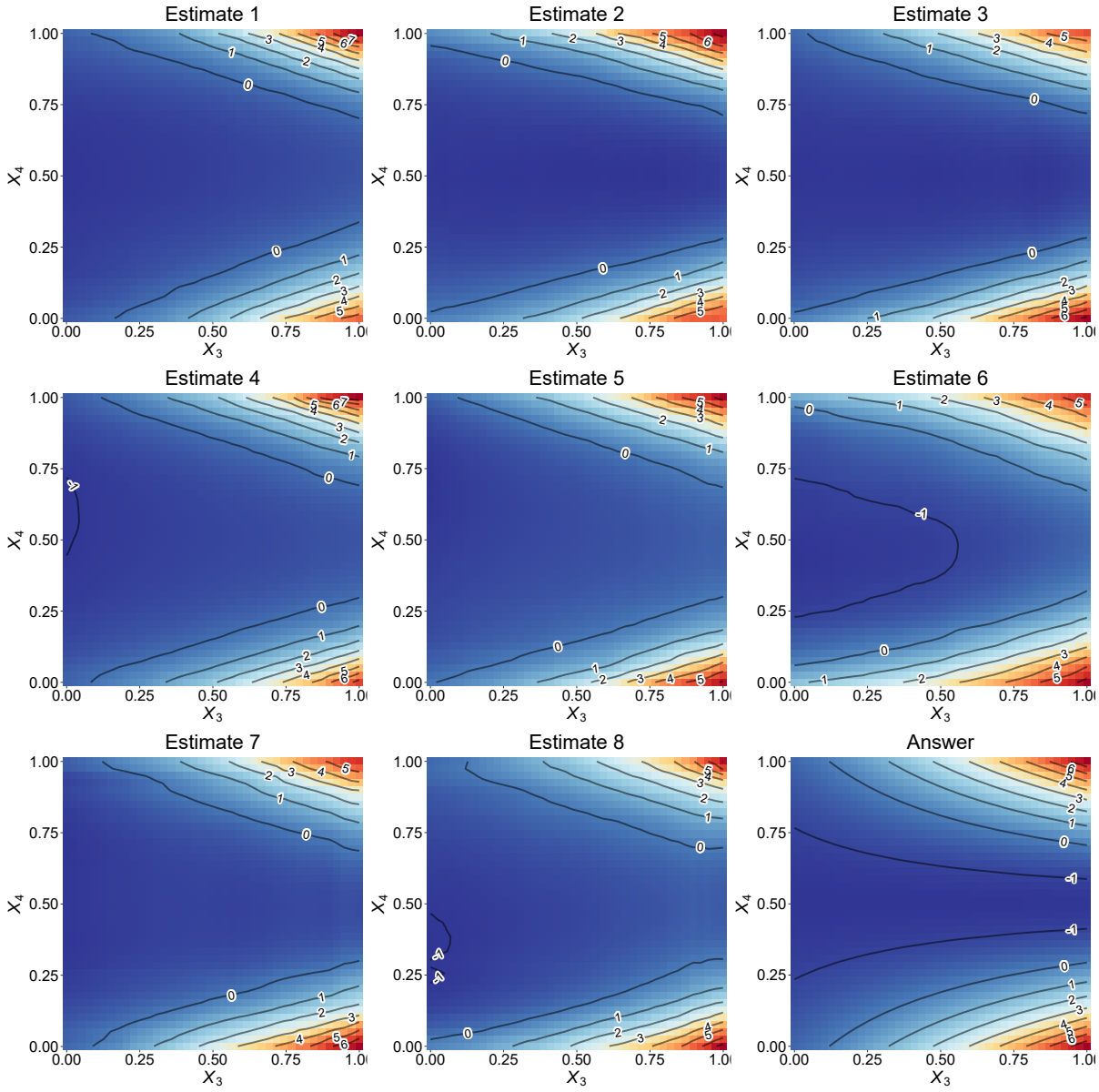


Figure A.8: Marginal joint effect of X_3 and X_4. Estimate 1–8 are posterior mean ALE joint effect $\hat{\bar{Q}}_{34}(\tau, x_3, x_4)$ of 8 replicates at quantile level $\tau = 0.05$. “Answer" represents the ground truth.

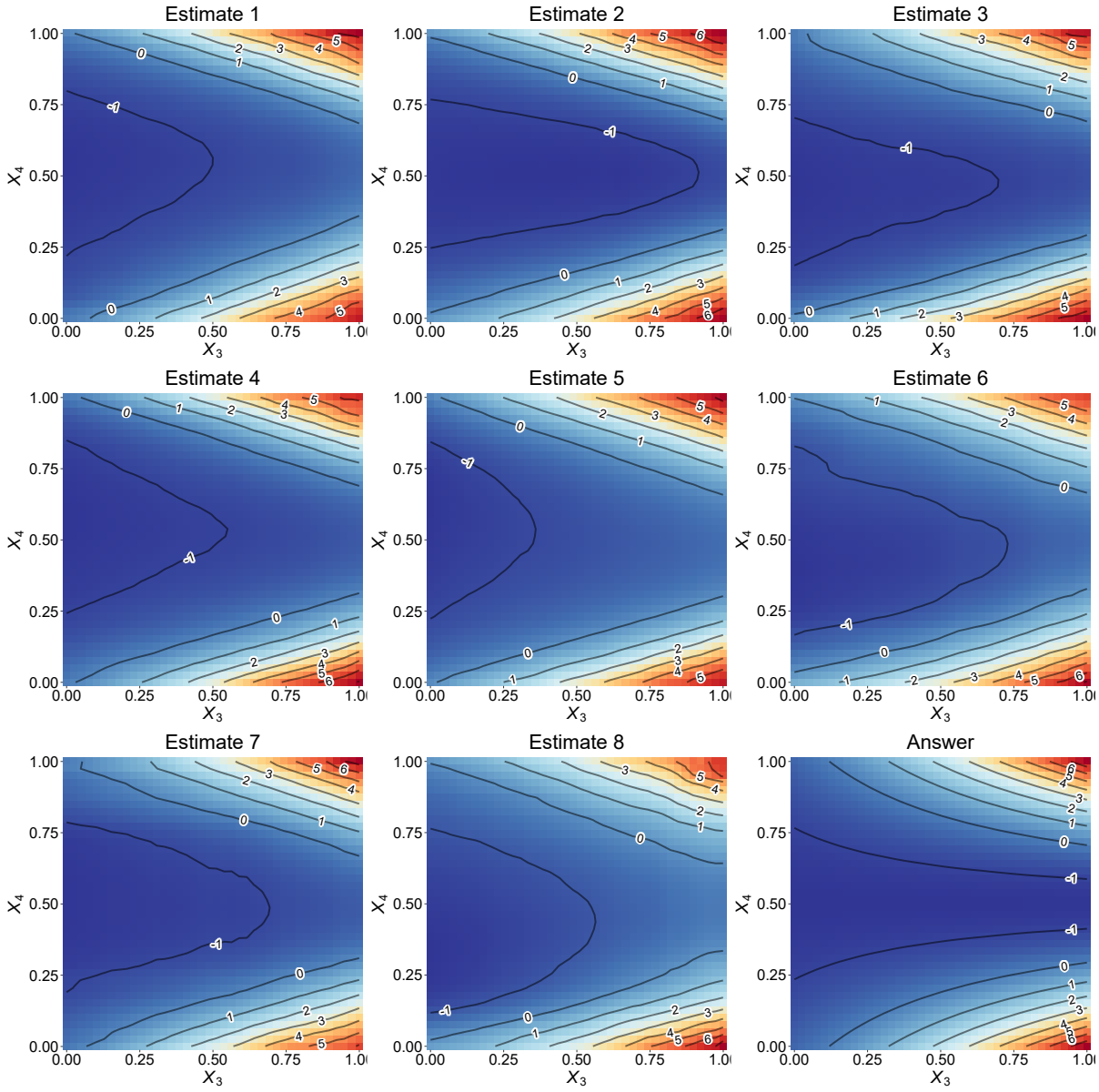


Figure A.9: Marginal joint effect between X_3 and X_4. Estimate 1–8 are posterior mean ALE joint effect $\hat{\bar{Q}}_{34}(\tau, x_3, x_4)$ of 8 replicates at quantile level $\tau = 0.5$. "Answer" represents the ground truth.

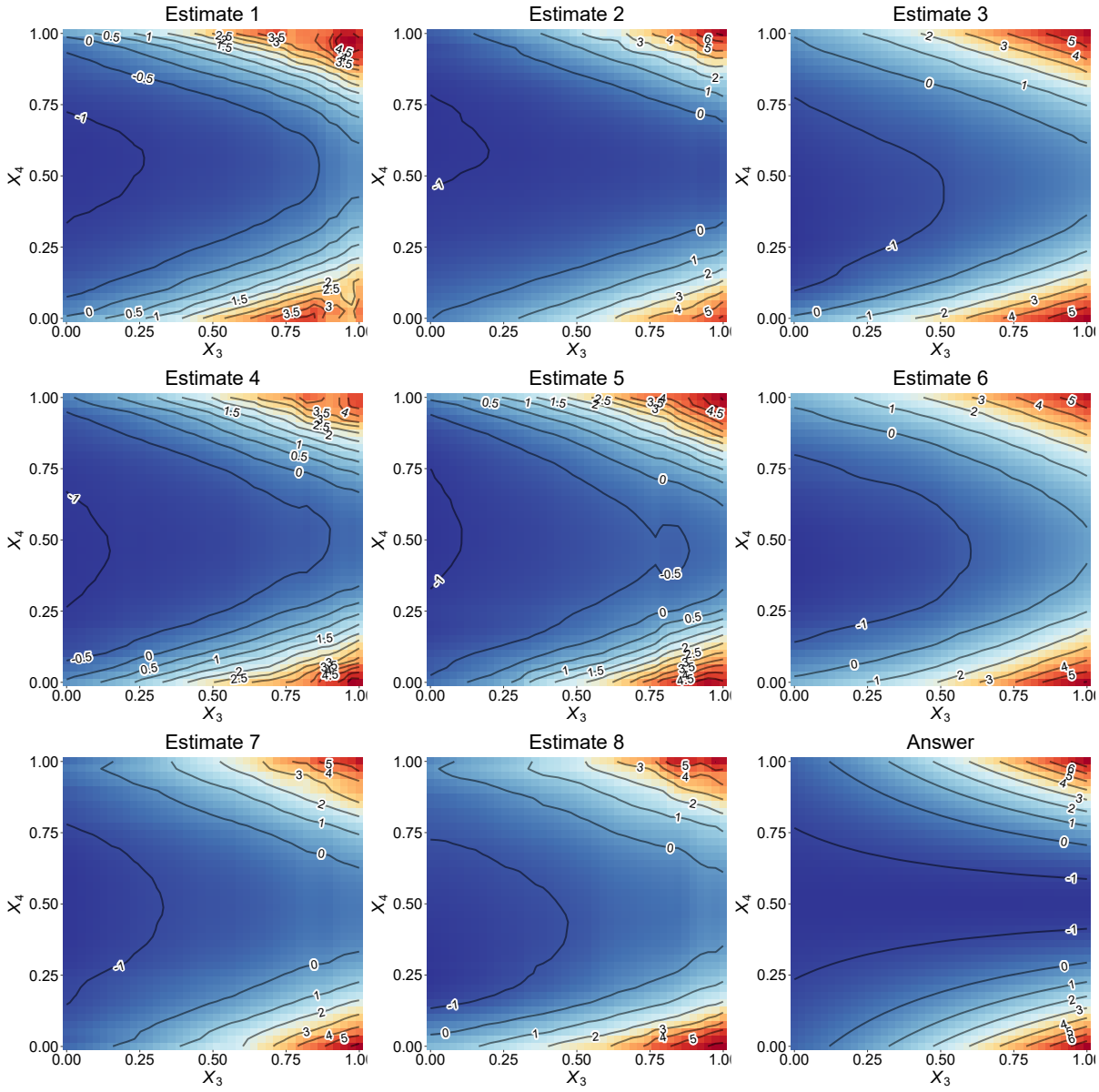


Figure A.10: Marginal joint effect of X_3 and X_4. Estimate 1–8 are posterior mean ALE joint effect $\hat{\bar{Q}}_{34}(\tau, x_3, x_4)$ of 8 replicates at quantile level $\tau = 0.95$. “Answer" represents the ground truth.

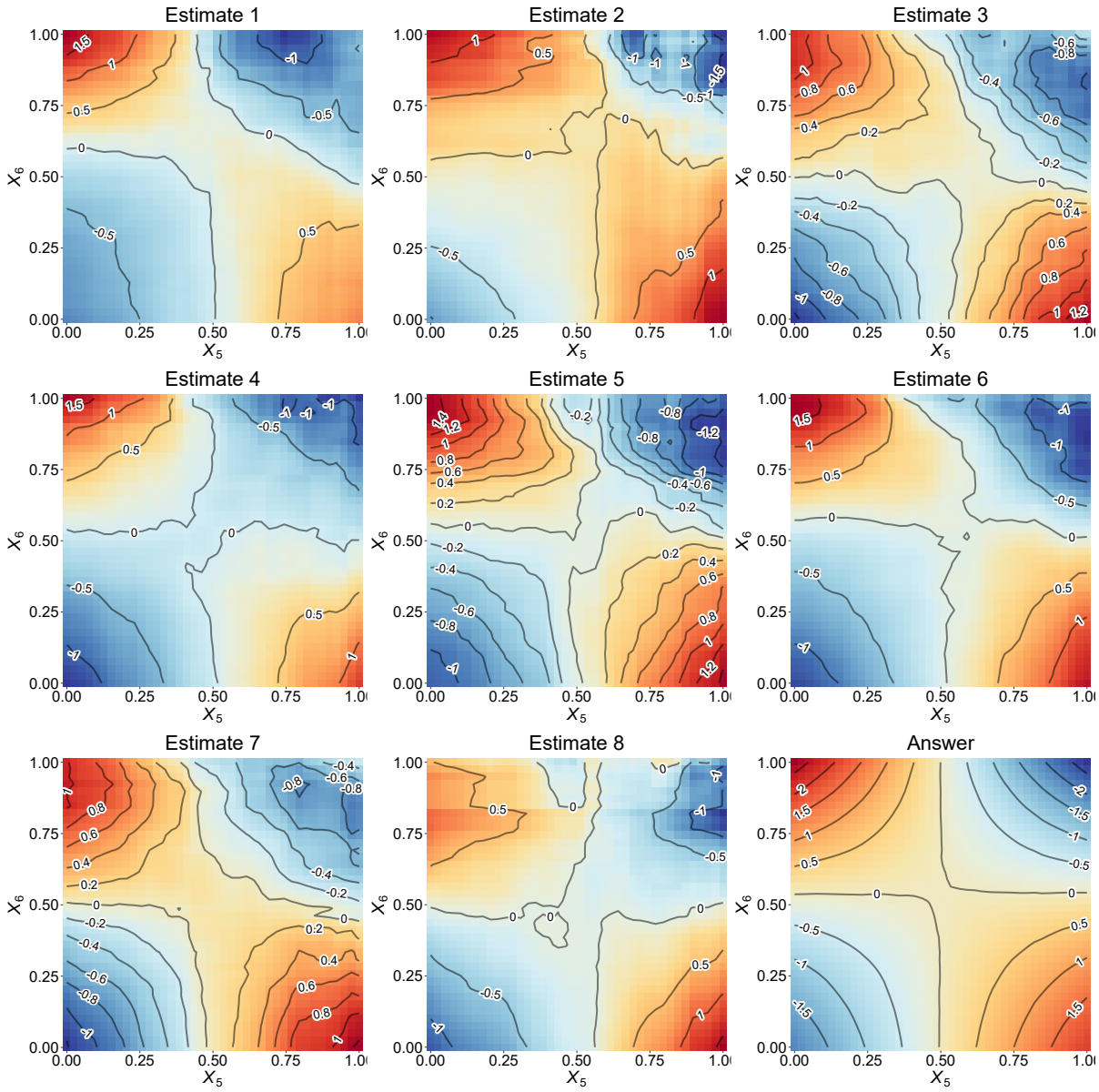


Figure A.11: Marginal interaction effect between X_5 and X_6. Estimate 1–8 are posterior mean ALE joint effect $\hat{\tilde{Q}}^I_{56}(\tau, x_5, x_6)$ of 8 replicates at quantile level $\tau = 0.05$. "Answer" represents the ground truth.

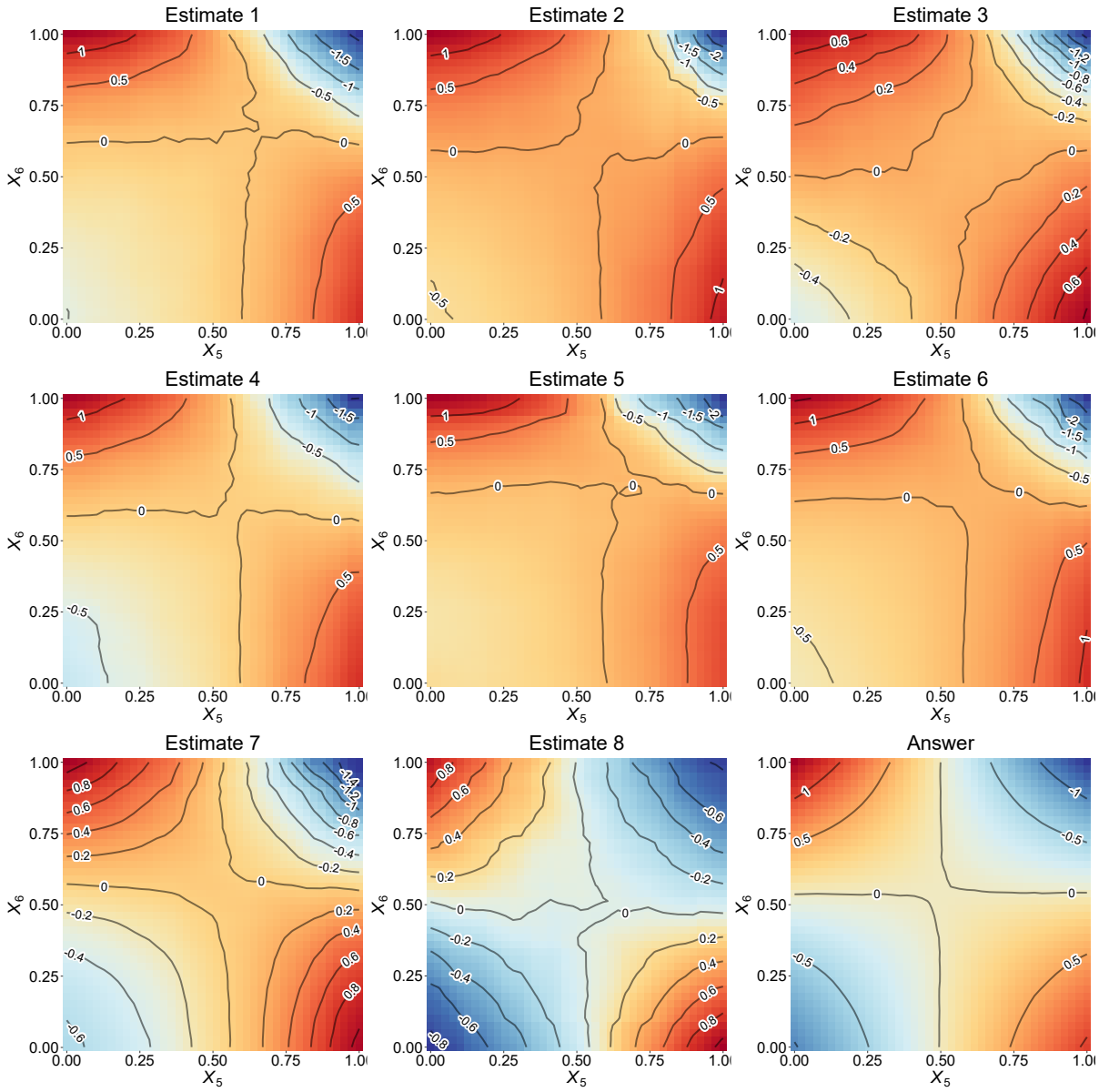


Figure A.12: Marginal interaction joint of X_5 and X_6. Estimate 1–8 are posterior mean ALE interaction effect $\hat{\bar{Q}}^I_{56}(\tau, x_5, x_6)$ of 8 replicates at quantile level $\tau = 0.5$. “Answer" represents the ground truth.

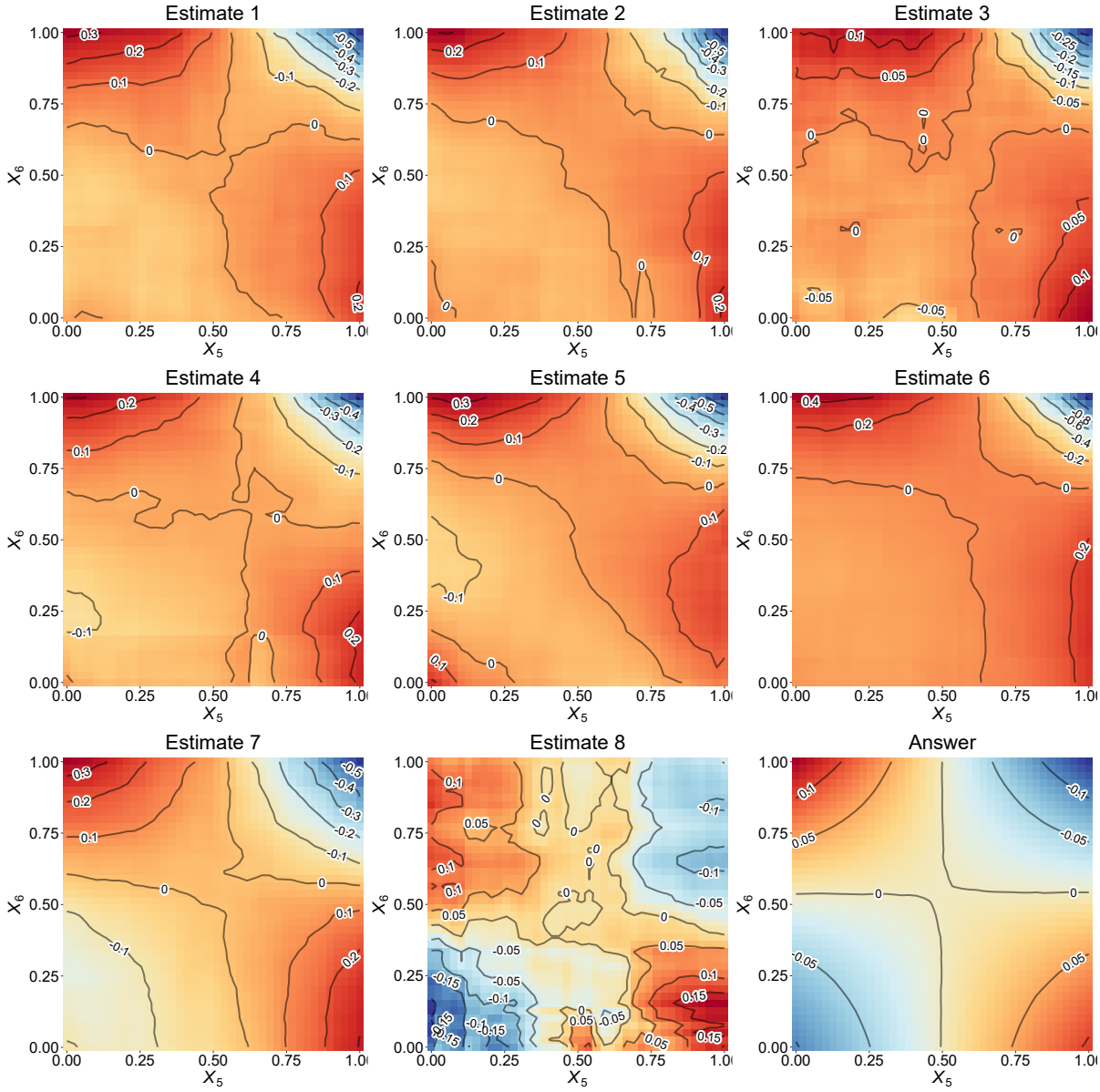


Figure A.13: Marginal interaction effect between X_5 and X_6. Estimate 1–8 are posterior mean ALE interaction effect $\hat{\bar{Q}}^I_{56}(\tau, x_5, x_6)$ of 8 replicates at quantile level $\tau = 0.95$. "Answer" represents the ground truth.

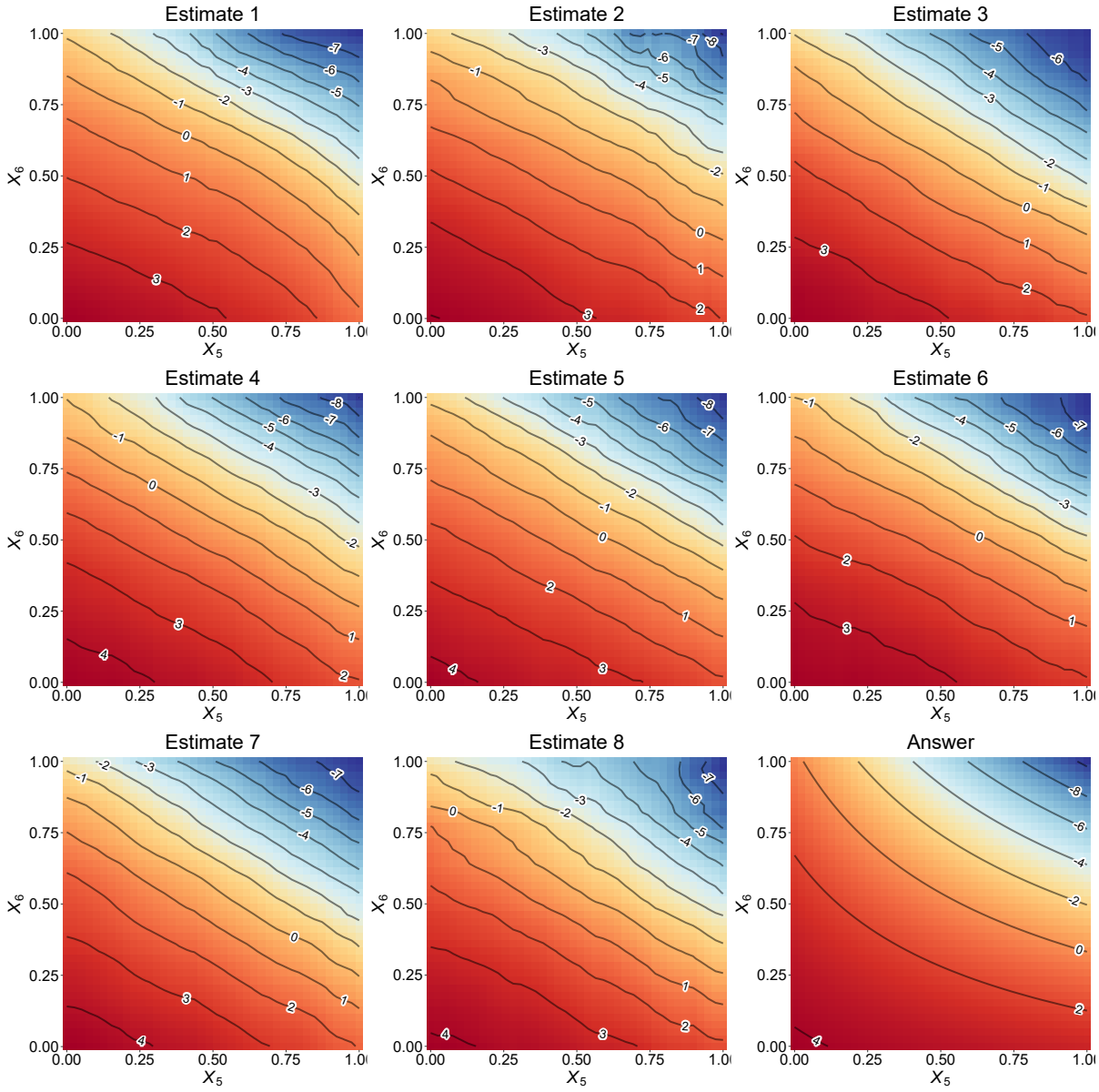


Figure A.14: Marginal joint effect of X_5 and X_6. Estimate 1–8 are posterior mean ALE joint effect $\hat{\bar{Q}}_{56}(\tau, x_5, x_6)$ of 8 replicates at quantile level $\tau = 0.05$. “Answer" represents the ground truth.

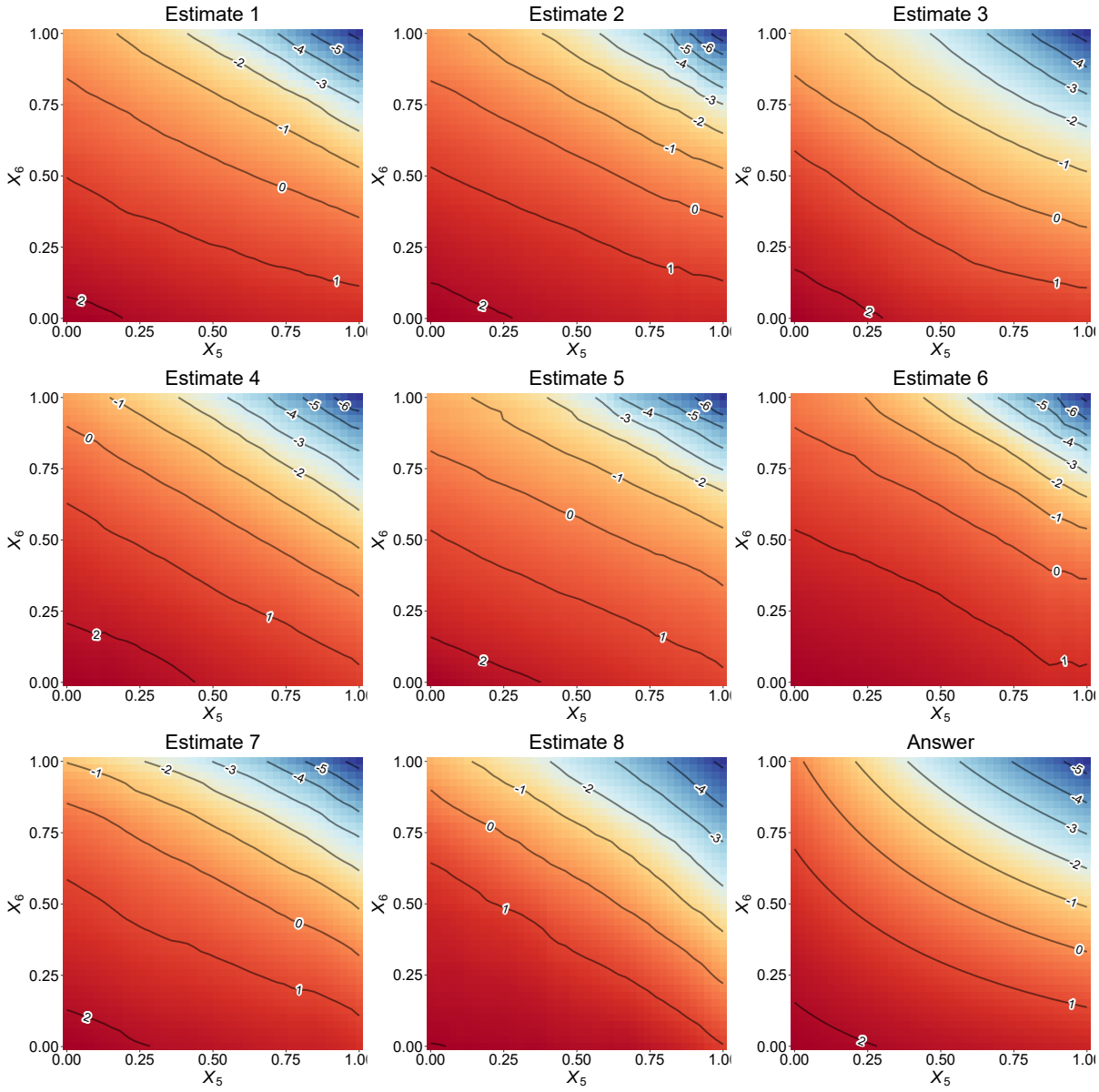


Figure A.15: Marginal joint effect of X_5 and X_6. Estimate 1–8 are posterior mean ALE joint effect $\hat{\bar{Q}}^{I}_{56}(\tau, x_5, x_6)$ of 8 replicates at quantile level $\tau = 0.5$. “Answer" represents the ground truth.

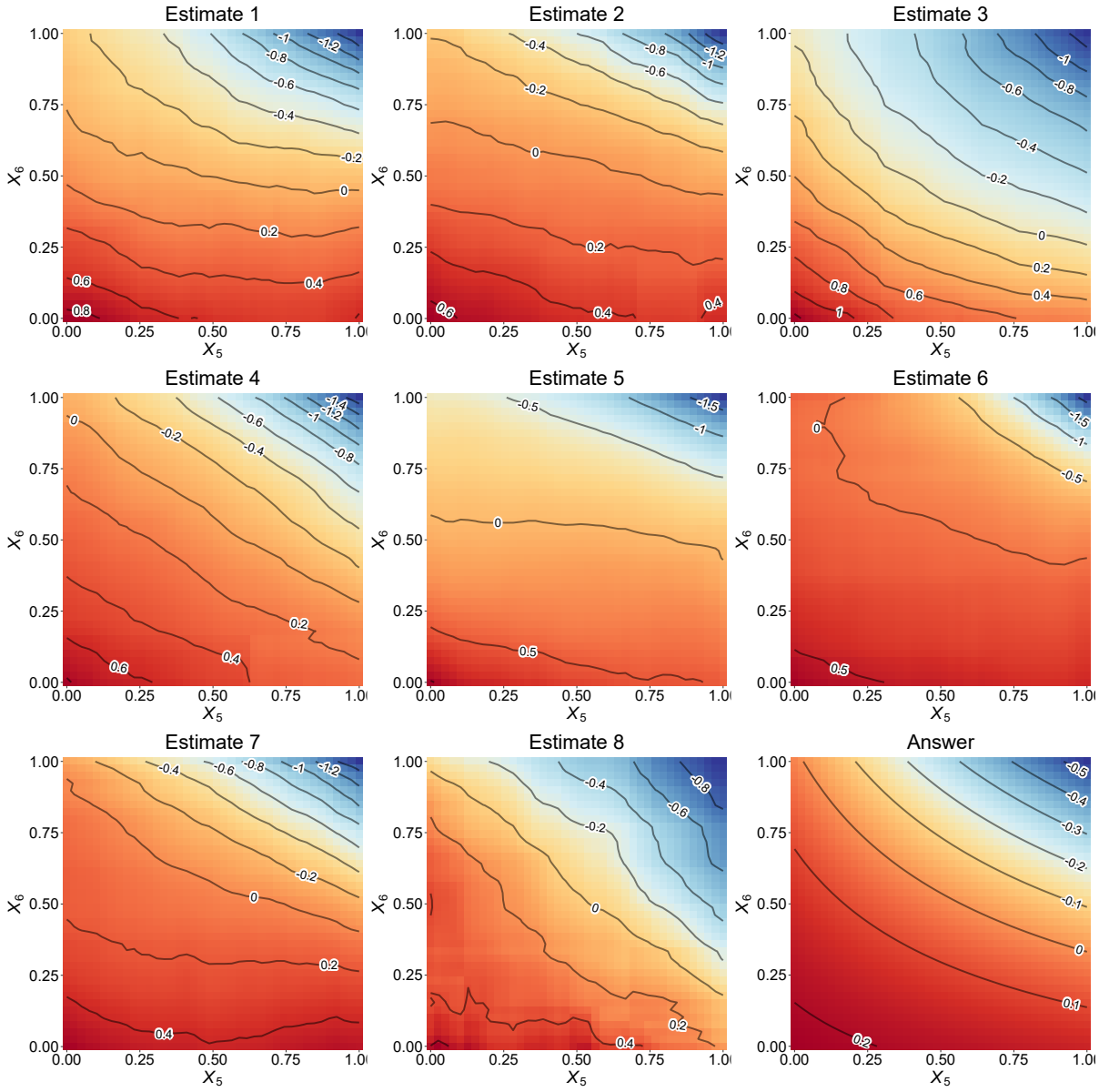


Figure A.16: Marginal joint effect of X_5 and X_6. Estimate 1–8 are posterior mean ALE joint effect $\hat{\bar{Q}}^{I}_{56}(\tau, x_5, x_6)$ of 8 replicates at quantile level $\tau = 0.95$. “Answer" represents the ground truth.

APPENDIX

B

SUPPLEMENT TO “A BAYESIAN SEMIPARAMETRIC METHOD FOR ESTIMATING CAUSAL QUANTILE EFFECTS"

B.1 Detail of Simulation 2

The exact form of the true model is

$$
\begin{aligned}
X_j &\sim \begin{cases} \mathscr{U}(0,1) & j=1,2,3 \\ \mathscr{U}(1,2) & j=4,5,6 \\ \mathscr{B}ern(0.5) & j=7,...,12 \end{cases} \\
T|\boldsymbol{X} &\sim \mathscr{B}ern(Z) \\
Y(0)|\boldsymbol{X} &\sim \sqrt{Z}\mathscr{N}\left(2Z^2+X_4+X_3,0.5^2\right)+\left(1-\sqrt{Z}\right)\mathscr{N}\left(Z^2+X_2-\sum_{j=1}^{3}X_j^2,0.8^2\right) \\
Y(1)|\boldsymbol{X} &\sim 0.6\mathscr{N}(-Z,0.8^2)+0.4\mathscr{N}(X_5+Z,1)
\end{aligned}
$$

where $Z=\text{expit}\left(-2.125+0.5X_1X_4+X_2X_5+\sum_{j=1}^{6}X_jX_{j+6}\right)$.

B.2 Detail of Simulation 3

The exact form of the true model is

$$
\begin{aligned}
\begin{pmatrix} X_1 \\ X_2 \\ X_3 \\ X_4 \end{pmatrix} &\sim \mathscr{N}\left[\begin{pmatrix} 0 \\ 0 \\ 0 \\ 0 \end{pmatrix}, \begin{pmatrix} 1 & 0.5 & 0.2 & 0.3 \\ 0.5 & 1 & 0.7 & 0 \\ 0.2 & 0.7 & 1 & 0 \\ 0.3 & 0 & 0 & 1 \end{pmatrix}\right] \\
T|\boldsymbol{X} &\sim \mathscr{B}ern\left(\text{expit}\left\{\boldsymbol{W}^o\times\tanh\left(\boldsymbol{W}^h\boldsymbol{X}^\top+\boldsymbol{b}^h\right)^\top+b^o\right\}\right) \\
Y(0)|\boldsymbol{X} &\sim \textit{Skew-}\mathscr{N}(2\tanh(X_2-X_3+0.5X_4),0.5,3) \\
Y(1)|\boldsymbol{X} &\sim \mathscr{N}\left(2\tanh\left(X_1+0.5X_2-X_3^2\right),0.5^2\right)
\end{aligned}
$$

where

$$\boldsymbol{W}^h = \begin{pmatrix} -0.99 & -1.1 & -0.14 & -0.26 \\ -0.18 & 0.03 & -1.45 & -0.07 \\ -0.44 & 0.19 & 0.86 & 0.36 \\ -1.07 & 0.67 & -0.58 & -0.13 \\ 0.12 & -0.37 & 0.47 & 1.25 \end{pmatrix}, \quad \boldsymbol{b}^h = \begin{pmatrix} 0.96 \\ 0.64 \\ 0.74 \\ -0.46 \\ 0.21 \end{pmatrix}$$

$$\boldsymbol{W}^o = \begin{pmatrix} -0.15 \\ 0.3 \\ -0.004 \\ -0.21 \\ -0.88 \end{pmatrix}, \quad b^o = -0.05.$$

APPENDIX

C

SUPPLEMENT TO "SINGLE-INDEX QUANTILE REGRESSION WITH MISSING AT RANDOM DATA"

C.1 Proof of Theorem 5.1

We first prove the convergence of the first element, $A_n(\boldsymbol{x})$. By definition, we have

$$\begin{aligned} A_n(\boldsymbol{x}) &= \frac{1}{n}\sum_{j=1}^{n}(1-\delta_j)\frac{\mathcal{K}_h(\boldsymbol{X}_j-\boldsymbol{x})}{\hat{r}_n(\boldsymbol{X}_j)} \\ &= \frac{1}{n}\sum_{j=1}^{n}\mathcal{K}_h(\boldsymbol{X}_j-\boldsymbol{x})\times\frac{\sum_{j=1}^{n}\frac{1-\delta_j}{\hat{r}_n(\boldsymbol{X}_j)}\mathcal{K}_h(\boldsymbol{X}_j-\boldsymbol{x})}{\sum_{j=1}^{n}\mathcal{K}_h(\boldsymbol{X}_j-\boldsymbol{x})} \\ &= \hat{f}_n(\boldsymbol{x})\hat{m}_n(\boldsymbol{x}) \end{aligned}$$

Now, by using the uniform convergence property of $\hat{m}_n(\boldsymbol{x})$ and $\hat{\pi}_n(\boldsymbol{x})$, we have

$$\hat{f}_n(\boldsymbol{x})\hat{m}_n(\boldsymbol{x}) \xrightarrow{\mathbb{P}} \hat{f}_n(\boldsymbol{x})\mathbb{E}\left\{\frac{1-\delta}{\hat{f}_n(\boldsymbol{x})\hat{\pi}_n(\boldsymbol{x})}\middle|\boldsymbol{X}=\boldsymbol{x}\right\} \xrightarrow{\mathbb{P}} \frac{1-\pi_0(\boldsymbol{x})}{\pi_0(\boldsymbol{x})}. \quad \square$$

We then prove the convergence of the second element, $B_n(\boldsymbol{x};\pi,\hat{\boldsymbol{\gamma}})$. By definition, we have

$$\begin{aligned}
B_n(\boldsymbol{x};\pi,\hat{\boldsymbol{\gamma}}) &= \frac{1}{n}\sum_{j=1}^{n}\left(1-\frac{\delta_j}{\pi(\boldsymbol{X}_j;\pi,\hat{\boldsymbol{\gamma}})}\right)\frac{\mathscr{K}_h(\boldsymbol{X}_j-\boldsymbol{x})}{\hat{r}_n(\boldsymbol{X}_j)} \\
&= \frac{1}{n}\sum_{j=1}^{n}\mathscr{K}_h(\boldsymbol{X}_j-\boldsymbol{x})\times\frac{\sum_{j=1}^{n}\frac{1-\delta_j/\pi(\boldsymbol{X}_j;\pi,\hat{\boldsymbol{\gamma}})}{\hat{r}_n(\boldsymbol{X}_j)}\mathscr{K}_h(\boldsymbol{X}_j-\boldsymbol{x})}{\sum_{j=1}^{n}\mathscr{K}_h(\boldsymbol{X}_j-\boldsymbol{x})} \\
&= \hat{f}_n(\boldsymbol{x})\hat{\tilde{m}}_n(\boldsymbol{x};\pi,\hat{\boldsymbol{\gamma}})
\end{aligned}$$

Now, by using the uniform convergence property of $\hat{\tilde{m}}_n(\boldsymbol{x};\pi,\hat{\boldsymbol{\gamma}})$ and $\hat{\pi}_n(\boldsymbol{x})$, we have

$$\hat{f}_n(\boldsymbol{x})\hat{\tilde{m}}_n(\boldsymbol{x};\pi,\hat{\boldsymbol{\gamma}}) \xrightarrow{\mathbb{P}} \hat{f}_n(\boldsymbol{x})\mathbb{E}\left\{\frac{1-\delta_i/\pi(\boldsymbol{x};\pi,\hat{\boldsymbol{\gamma}})}{\hat{f}_n(\boldsymbol{x})\hat{\pi}_n(\boldsymbol{x})}\middle|\boldsymbol{X}=\boldsymbol{x}\right\} \xrightarrow{\mathbb{P}} \frac{1}{\pi_0(\boldsymbol{x})}-\frac{1}{\pi(\boldsymbol{x};\pi,\hat{\boldsymbol{\gamma}})}. \quad \square$$