Smoothed Quantile Regression with Large-Scale Inference

Xuming He[*], Xiaoou Pan[†], Kean Ming Tan[‡] and Wen-Xin Zhou[§]

Abstract

Quantile regression is a powerful tool for learning the relationship between a scalar response and a multivariate predictor in the presence of heavier tails and/or data heterogeneity. In the present paper, we consider statistical inference for quantile regression with large-scale data in the "increasing dimension" regime. We provide a comprehensive study of a convolution-type smoothing approach to achieving an adequate approximation to computation and inference for quantile regression. The ensuing estimator, which we refer to as *conquer*, turns the non-differentiable quantile loss function into a twice-differentiable, globally convex, and locally strongly convex surrogate, which admits a fast and scalable Barzilai-Borwein gradient-based algorithm to perform optimization, and a Rademacher multiplier bootstrap method for statistical inference. In the theoretical investigations of the conquer estimator, we establish nonasymptotic error bounds on the Bahadur-Kiefer linearization, from which we show that the asymptotic normality of the smoothed quantile regression estimator holds under a weaker requirement on the dimension of the predictors than needed for the exact quantile regression estimator. Our numerical studies confirm the conquer estimator as a practical and reliable approach to large-scale inference for quantile regression.

Keywords: Bahadur-Kiefer representation; convolution; gradient descent; multiplier bootstrap.

[*]Department of Statistics, University of Michigan, Ann Arbor, MI 48109, USA. E-mail:xmhe@umich.edu.

[†]Department of Mathematics, University of California, San Diego, La Jolla, CA, 92093, USA. E-mail:xip024@ucsd.edu.

[‡]Department of Statistics, University of Michigan, Ann Arbor, MI 48109, USA. E-mail:keanming@umich.edu.

[§]Department of Mathematics, University of California, San Diego, La Jolla, CA 92093, USA. E-mail:wez243@ucsd.edu.

1 Introduction

Quantile regression (QR) is a useful statistical tool for modeling and inferring the relationship between a scalar response y and a p-dimensional predictor $\boldsymbol{x}$ (Koenker and Bassett, 1978). Compared to the least squares regression that focuses on modeling the conditional mean of y given $\boldsymbol{x}$, QR allows modeling of the entire conditional distribution of y given $\boldsymbol{x}$, and thus provides valuable insights into heterogeneity in the relationship between $\boldsymbol{x}$ and y. Moreover, quantile regression is robust against outliers and can be performed for skewed or heavy-tailed response distributions without a correct specification of the likelihood. These advantages make QR an appealing method to explore data features that are invisible to the least squares regression. We refer the reader to Koenker (2005) and Koenker et al. (2017) for an extensive overview of QR in terms of methods, theory, computation, and various extensions under complex data structures.

Quantile regression involves a convex optimization problem with a piecewise linear loss function, also known as the *check function* and *pinball loss*. One can formulate QR as a linear programming problem, solvable by the interior point method with a computational complexity of $O(n^{1+\alpha}p^3 \log n)$ with $\alpha \in (0, 1/2)$, where n is the sample size and p is the parametric dimension. When n is large relative to p, an efficient algorithm based on pre-processing has an improved complexity of $O\{(np)^{2(1+\alpha)/3}p^3 \log n + np\}$, and thus can be more efficient than solving a least squares problem when p is small (Portnoy and Koenker, 1997). However, primarily due to the non-differentiability and lack of strong convexity of the loss function, QR remains computationally expensive for large-scale data when both n and p are large. We refer to Chapter 5 of Koenker et al. (2017) for an overview of prevailing computational methods for quantile regression, such as simplex-based algorithms (Barrodale and Roberts, 1974; Koenker and d'Orey, 1987), interior point methods (Portnoy and Koenker, 1997), and alternating direction method of multipliers among other first-order proximal methods (Parikh and Boyd, 2014).

We consider conducting large-scale inference for quantile regression under the setting in which p is large and n is even larger. Two general principles have been widely used to suit this purpose. The first uses a nonparametric estimate of the asymptotic variance (Gutenbrunner and Jurečková, 1992) that involves the conditional density of the response given the covariates, yet such an estimate can be fairly unstable. Even if the asymptotic variance is well estimated, its approximation accuracy to the

finite-sample variance depends on the design matrix and the quantile level. Resampling methods, on the other hand, provide a more reliable approach to inference for QR under a wide variety of settings (Parzen, Wei and Ying, 1994; He and Hu, 2002; Kocherginsky, He and Hu, 2005; Feng, He and Hu, 2011). Inevitably, the resampling approach requires repeatedly computing QR estimates up to thousands of times, and therefore is unduly expensive for large-scale data.

Theoretically, valid statistical inference is often justified by asymptotic normal approximations to QR estimators. The Bahadur-Kiefer representation of the nonlinear QR estimators are essential to this end, as shown in Arcones (1996) and He and Shao (1996). In large-p (non)asymptotic settings in which the parametric dimension p may tend to infinity with the sample size, we refer to Welsh (1989), He and Shao (2000), and Pan and Zhou (2020) for normal approximation results of the QR estimators under fixed and random designs. The question of how large p can be relative to n to ensure asymptotic normality has been addressed by those authors. It is now recognized that we may have to pay a price here as compared to M-estimators with smooth loss functions that are at least continuously twice differentiable.

To circumvent the non-differentiability of the QR loss function, Horowitz (1998) proposed to smooth the indicator part of the check function via the survival function of a kernel. This smoothing method, which we refer to as *Horowitz's smoothing* throughout, has been widely used for various QR-related problems (Whang, 2006; Kaplan and Sun, 2017; Galvao and Kato, 2016; Wang, Stefanski and Zhu, 2012; Wu, Ma and Yin, 2015; de Castro et al., 2019; Chen, Liu and Zhang, 2019). However, Horowitz's smoothing gains smoothness at the cost of convexity, which inevitably raises optimization issues. In general, computing a global minimum of a non-convex function is intractable: finding an ϵ-suboptimal point for a k-times continuously differentiable function $f : \mathbb{R}^p \to \mathbb{R}$ requires at least as many as $(1/\epsilon)^{p/k}$ evaluations of the function and its first k derivatives (Nemirovski and Yudin, 1983). As we shall see from the numerical studies in Section 5, the convergence of gradient-based algorithms can be relatively slow for high and low quantile levels. To address the aforementioned issue, Fernandes, Guerre and Horta (2019) proposed a convolution-type smoothing method that yields a convex and twice differentiable loss function, and studied the asymptotic properties of the smoothed estimator when p is fixed. To distinguish this approach from Horowitz's smoothing, we adopt the term *conquer* for convolution-type smoothed quantile regression.

In this paper, we first provide an in-depth statistical analysis of conquer under various array

(non)asymptotic settings in which p increases with n. Our results reveal a key feature of the smoothing parameter, often referred to as the bandwidth: the bandwidth adapts to both the sample size n and dimensionality p, so as to achieve a tradeoff between statistical accuracy and computational stability. Since the convolution smoothed loss function is globally convex and locally strongly convex, we propose an efficient gradient descent algorithm with the Barzilai-Borwein stepsize and a Huber-type initialization. The proposed algorithm is implemented via `RcppArmadillo` (Eddelbuettel and Sanderson, 2014) in the R package `conquer`. We next focus on large-scale statistical inference (hypothesis testing and confidence estimation) with large p and large n. We propose a bootstrapped conquer method that has reduced computational complexity when the conquer estimator is used as initialization. Under appropriate restrictions on dimension, we establish the consistency (or concentration), Bahadur representation, asymptotic normality of the conquer estimator as well as the validity of the bootstrap approximation. In the following, we provide more details on the computational and statistical contributions of this paper.

Theoretically, by allowing p to grow with n, the 'complexity' of the function classes that we come across in the analysis also increases with n. Conventional asymptotic tools for proving the bootstrap validity are based on weak convergence arguments (van der Vaart and Wellner, 1996), which are not directly applicable in the finite-sample setting. In this paper we turn to a more refined and self-contained analysis, and prove a new local restricted strong convexity (RSC) property for the empirical smoothed quantile loss. This validates the key merit of convolution-type smoothing, i.e., local strong convexity. The smoothing method involves a bandwidth, denoted by h. Theoretically, we show that with sub-exponential random covariates (relaxing the bounded covariates assumption in Fernandes, Guerre and Horta (2019)), conquer exhibits an ℓ_2-error in the order of $\sqrt{(p+t)/n} + h^2$ with probability at least $1 - 2e^{-t}$. When h is of order $\{(p+t)/n\}^{\gamma}$ for any $\gamma \in [1/4, 1/2]$, the conquer estimation is first-order equivalent to QR. Under slightly more stringent sub-Gaussian condition on the covariates, we show that the Bahadur-Kiefer linearization error of conquer is of order $(p + t)/(nh^{1/2}) + h\sqrt{(p+t)/n} + h^3$ with probability at least $1 - 3e^{-t}$. Based on such a representation, we establish a Berry-Esseen bound for linear functionals of conquer, which lays the theoretical foundation for testing general linear hypotheses, encompassing covariate-effect analysis, analysis of variance, and model comparisons, to name a few. It is worth noting that with a properly chosen h, the linear functional of conquer is asymptotically normal as long as $p^{8/3}/n \to 0$, which improves

the best known growth condition on p for standard QR (Welsh, 1989; He and Shao, 2000; Pan and Zhou, 2020). We attribute this gain to the effect of smoothing. Under similar conditions, we further establish upper bounds on both estimation and Bahadur-Kiefer linearization errors for the bootstrapped conquer estimator.

In the context of nonparametric density or regression estimation, it is known that when higher-order kernels are used (and if the density or regression function has enough derivatives), the bias is proportional to h^{ν} for some $\nu \geq 4$ which is of better order than h^2. While a higher-order kernel has negative parts, the resulting smoothed loss is non-convex and thus brings the computational issue once again. Motivated by the two-stage procedure proposed by Bickel (1975) whose original idea is to improve an initial estimator that is already consistent but not efficient, we further propose a one-step conquer estimator using higher-order kernels but without the need for solving a large-scale non-convex optimization. With increasing degrees of smoothness, the one-step conquer is asymptotically normal under a milder dimension constraint of roughly $p^2/n \to 0$.

To better appreciate the computational feasibility of conquer for large-scale problems, we compare it with standard QR on large synthetic datasets, where the latter is implemented by the R package `quantreg` (Koenker, 2019) using the Frisch-Newton approach after preprocessing "pfn". We generate independent data vectors $\{y_i, \boldsymbol{x}_i\}_{i=1}^n$ from a linear model $y_i = \beta_0^* + \langle \boldsymbol{x}_i, \boldsymbol{\beta}^* \rangle + \varepsilon_i$, where $(\beta_0^*, \boldsymbol{\beta}^{*\mathrm{T}})^{\mathrm{T}} = (1, \ldots, 1)^{\mathrm{T}} \in \mathbb{R}^{p+1}$, $\boldsymbol{x}_i \sim \mathcal{N}_p(0, \mathbf{I})$, and the independent errors $\varepsilon_i \sim t_2$, for $i = 1, 2, \ldots, n$. We report the estimation error and elapsed time for increasing sample sizes $n \in \{1000, 5000, 10000, \ldots, 100000\}$ and dimension $p = \lfloor n^{1/2} \rfloor$, the largest integer that is less than or equal to $n^{1/2}$. Figure 1 displays the average estimation error and average elapsed time based on 20 Monte Carlo samples. This experiment shows promise of conquer as a practically useful tool for large-scale quantile regression analysis. More empirical evidence will be given in the latter section.

The rest of the paper is organized as follows. We start with a brief review of linear quantile regression and the convolution-type smoothing method in Section 2. Explicit forms of the smoothed check functions are provided for several representative kernel functions in nonparametric statistics. We introduce the multiplier bootstrap for statistical inference in Section 2.4. In Section 3, we provide a comprehensive theoretical study of conquer from a nonasymptotic viewpoint, which directly leads to array asymptotic results. Specifically, the bias incurred by smoothing the quantile loss is characterized in Section 3.1. In Section 3.2, we establish the rate of convergence, Bahadur-Kiefer
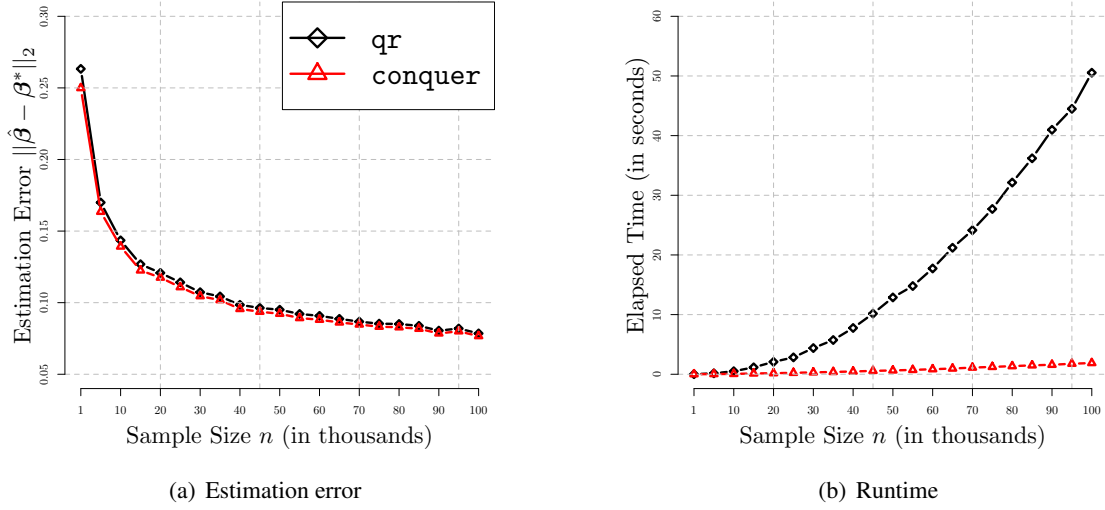
(a) Estimation error	(b) Runtime

Figure 1: A numerical comparison between conquer and QR. The latter is implemented by the R package `quantreg` using the "pfn" method. Panels (a) and (b) display, respectively, the "estimation error versus sample size" and "elapsed time versus sample size" as the size of the problem increases.

representation, and Berry-Esseen bound for conquer in a large-p and larger-n regime. Results for its bootstrap counterpart are provided in Section 3.3. A Barzilai-Borwein gradient-based algorithm with a Huber-type warm start is detailed in Section 4. We conclude the paper with an extensive numerical study in Section 5 to illustrate the finite-sample performance of conquer in large-scale quantile regression analysis. We defer the proofs of all theoretical results as well as the full details of the one-step conquer to online supplementary materials.

NOTATION: For every integer $k \geq 1$, we use $\mathbb{R}^k$ to denote the the k-dimensional Euclidean space. The inner product of any two vectors $\boldsymbol{u} = (u_1, \ldots, u_k)^{\mathrm{T}}, \boldsymbol{v} = (v_1, \ldots, v_k)^{\mathrm{T}} \in \mathbb{R}^k$ is defined by $\boldsymbol{u}^{\mathrm{T}}\boldsymbol{v} = \langle \boldsymbol{u}, \boldsymbol{v} \rangle = \sum_{i=1}^{k} u_i v_i$. We use $\|\cdot\|_p$ ($1 \leq p \leq \infty$) to denote the ℓ_p-norm in $\mathbb{R}^k$: $\|\boldsymbol{u}\|_p = (\sum_{i=1}^{k} |u_i|^p)^{1/p}$ and $\|\boldsymbol{u}\|_\infty = \max_{1 \leq i \leq k} |u_i|$. Throughout this paper, we use bold capital letters to represent matrices. For $k \geq 2$, $\mathbf{I}_k$ represents the identity matrix of size k. For any $k \times k$ symmetric matrix $\mathbf{A} \in \mathbb{R}^{k \times k}$, $\|\mathbf{A}\|_2$ denotes the operator norm of $\mathbf{A}$. If $\mathbf{A}$ is positive semidefinite, we use $\|\cdot\|_{\mathbf{A}}$ to denote the vector norm linked to $\mathbf{A}$ given by $\|\boldsymbol{u}\|_{\mathbf{A}} = \|\mathbf{A}^{1/2}\boldsymbol{u}\|_2$, $\boldsymbol{u} \in \mathbb{R}^k$. For $r \geq 0$, define the Euclidean ball and sphere in $\mathbb{R}^k$ as $\mathbb{B}^k(r) = \{\boldsymbol{u} \in \mathbb{R}^k : \|\boldsymbol{u}\|_2 \leq r\}$ and $\mathbb{S}^{k-1}(r) = \partial \mathbb{B}^k(r) = \{\boldsymbol{u} \in \mathbb{R}^k : \|\boldsymbol{u}\|_2 = r\}$, respectively. For two sequences of non-negative numbers $\{a_n\}_{n \geq 1}$ and $\{b_n\}_{n \geq 1}$, $a_n \lesssim b_n$ indicates that there exists a constant $C > 0$ independent of n such that $a_n \leq C b_n$; $a_n \gtrsim b_n$ is equivalent to $b_n \lesssim a_n$; $a_n \asymp b_n$ is equivalent to $a_n \lesssim b_n$ and $b_n \lesssim a_n$.

2 Smoothed Quantile Regression

2.1 The linear quantile regression model

Given a univariate response variable $y \in \mathbb{R}$ and a p-dimensional covariate vector $\boldsymbol{x} = (x_1, \ldots, x_p)^\mathrm{T} \in \mathbb{R}^p$ with $x_1 \equiv 1$, the primary goal here is to learn the effect of $\boldsymbol{x}$ on the distribution of y. Let $F_{y|\boldsymbol{x}}(\cdot)$ be the conditional distribution function of y given $\boldsymbol{x}$. The dependence between y and $\boldsymbol{x}$ is then fully characterized by the conditional quantile functions of y given $\boldsymbol{x}$, denoted as $F_{y|\boldsymbol{x}}^{-1}(\tau)$, for $0 < \tau < 1$. We consider a linear quantile regression model at a given $\tau \in (0, 1)$, that is, the τ-th conditional quantile function is

$$F_{y|\boldsymbol{x}}^{-1}(\tau) = \langle \boldsymbol{x}, \boldsymbol{\beta}^*(\tau) \rangle, \tag{2.1}$$

where $\boldsymbol{\beta}^*(\tau) = (\beta_1^*(\tau), \ldots, \beta_p^*(\tau))^\mathrm{T} \in \mathbb{R}^p$ is the true quantile regression coefficient.

Let $\{(y_i, \boldsymbol{x}_i)\}_{i=1}^n$ be a random sample from $(y, \boldsymbol{x})$. The standard quantile regression estimator (Koenker and Bassett, 1978) is then given as

$$\widehat{\boldsymbol{\beta}}(\tau) \in \min_{\boldsymbol{\beta} \in \mathbb{R}^p} \widehat{Q}(\boldsymbol{\beta}) = \min_{\boldsymbol{\beta} \in \mathbb{R}^p} \frac{1}{n} \sum_{i=1}^n \rho_\tau(y_i - \langle \boldsymbol{x}_i, \boldsymbol{\beta} \rangle), \tag{2.2}$$

where

$$\rho_\tau(u) = u\{\tau - \mathbb{1}(u < 0)\} \tag{2.3}$$

is the τ-quantile loss function, also referred to as the pinball loss and check function. Statistical properties of $\widehat{\boldsymbol{\beta}}(\tau)$ have been extensively studied. We refer the reader to Koenker (2005) and Koenker et al. (2017) for more details.

2.2 Convolution-type smoothing

We consider the convolution-type smoothing approach proposed by Fernandes, Guerre and Horta (2019). For every $\boldsymbol{\beta} \in \mathbb{R}^p$, let $\widehat{F}(\cdot; \boldsymbol{\beta})$ be the empirical distribution function of the residuals $\{r_i(\boldsymbol{\beta}) := y_i - \langle \boldsymbol{x}_i, \boldsymbol{\beta} \rangle\}_{i=1}^n$, i.e., $\widehat{F}(t; \boldsymbol{\beta}) = (1/n) \sum_{i=1}^n \mathbb{1}\{r_i(\boldsymbol{\beta}) \leq t\}$. Given a fixed quantile level τ, the empirical

quantile loss $\widehat{Q}(\cdot)$ in (2.2) can be represented as

$$\widehat{Q}(\boldsymbol{\beta}) = \int_{-\infty}^{\infty} \rho_{\tau}(u)\, \mathrm{d}\widehat{F}(u; \boldsymbol{\beta}). \tag{2.4}$$

Given a bandwidth $h = h_n > 0$, let $\widehat{F}_h(\cdot; \boldsymbol{\beta})$ be the distribution function of the classical Rosenblatt–Parzen kernel density estimator:

$$\widehat{F}_h(u; \boldsymbol{\beta}) = \int_{-\infty}^{u} \widehat{f}_h(t; \boldsymbol{\beta})\, \mathrm{d}t \quad \text{with} \quad \widehat{f}_h(t; \boldsymbol{\beta}) = \frac{1}{nh} \sum_{i=1}^{n} K\left(\frac{t - r_i(\boldsymbol{\beta})}{h}\right), \tag{2.5}$$

where $K : \mathbb{R} \to \mathbb{R}^+ := [0, \infty)$ is a symmetric kernel that integrates to one. Replacing the empirical distribution function in (2.4) with $\widehat{F}_h$ yields the following smoothed empirical quantile loss

$$\widehat{Q}_h(\boldsymbol{\beta}) := \int_{-\infty}^{\infty} \rho_{\tau}(u)\, \mathrm{d}\widehat{F}_h(u; \boldsymbol{\beta}) = \frac{1}{nh} \sum_{i=1}^{n} \int_{-\infty}^{\infty} \rho_{\tau}(u) K\left(\frac{u + \langle \boldsymbol{x}_i, \boldsymbol{\beta}\rangle - y_i}{h}\right) \mathrm{d}u. \tag{2.6}$$

Accordingly, the smoothed QR estimator is given by

$$\widehat{\boldsymbol{\beta}}_h = \widehat{\boldsymbol{\beta}}_h(\tau) \in \operatorname*{argmin}_{\boldsymbol{\beta} \in \mathbb{R}^p} \widehat{Q}_h(\boldsymbol{\beta}). \tag{2.7}$$

As we shall see later, the ideal choice of bandwidth should adapt to the sample size n and dimension p, while the quantile level τ is prespecified and fixed. Thus, the dependence of $\widehat{\boldsymbol{\beta}}_h$ and $\widehat{Q}_h(\cdot)$ on τ will be assumed without display.

Given a kernel function $K(\cdot)$ and bandwidth $h > 0$, the smoothed quantile loss $\widehat{Q}_h(\cdot)$ defined in (2.6) can be equivalently written as $\widehat{Q}_h(\boldsymbol{\beta}) = (1/n) \sum_{i=1}^{n} \ell_h(y_i - \langle \boldsymbol{x}_i, \boldsymbol{\beta}\rangle)$, where

$$\ell_h(u) = (\rho_{\tau} * K_h)(u) = \int_{-\infty}^{\infty} \rho_{\tau}(v) K_h(v - u)\, \mathrm{d}v \tag{2.8}$$

with $*$ denoting the convolution operator and $K_h(u) = (1/h)K(u/h)$. Therefore, as stated in the Introduction, we refer to the aforementioned smoothing method as *conquer* throughout the paper. Commonly used kernel functions include: (a) uniform kernel $K(u) = (1/2)\mathbb{1}(|u| \le 1)$, (b) Gaussian kernel $K(u) = \phi(u) := (2\pi)^{-1/2}e^{-u^2/2}$, (c) logistic kernel $K(u) = e^{-u}/(1 + e^{-u})^2$, (d) Epanechnikov kernel $K(u) = (3/4)(1 - u^2)\mathbb{1}(|u| \le 1)$, and (e) triangular kernel $K(u) = (1 - |u|)\mathbb{1}(|u| \le 1)$. Explicit expressions of the corresponding smoothed loss function $\rho_{\tau} * K_h$ will be given in Section 4.

The convolution-type kernel smoothing yields an objective function $\beta \mapsto \widehat{Q}_h(\beta)$ that is twice continuously differentiable with gradient and hessian matrix

$$\nabla \widehat{Q}_h(\beta) = \frac{1}{n} \sum_{i=1}^{n} \{\overline{K}(-r_i(\beta)/h) - \tau\} x_i \quad \text{and} \quad \nabla^2 \widehat{Q}_h(\beta) = \frac{1}{n} \sum_{i=1}^{n} K_h(r_i(\beta)) x_i x_i^{\mathsf{T}}, \qquad (2.9)$$

respectively, where $\overline{K}(u) := \int_{-\infty}^{u} K(t)\, dt$. Provided that K is non-negative, $\widehat{Q}_h(\cdot)$ is a convex function for any $h > 0$, and $\widehat{\beta}_h = \widehat{\beta}_h(\tau)$ satisfies the first-order condition

$$\nabla \widehat{Q}_h(\widehat{\beta}_h) = \frac{1}{n} \sum_{i=1}^{n} \left\{ \overline{K}\left(\frac{\langle x_i, \widehat{\beta}_h \rangle - y_i}{h} \right) - \tau \right\} x_i = \mathbf{0}.$$

The smoothness and convexity of $\widehat{Q}_h$ warrant the superior computation efficiency of first-order gradient based algorithms for solving large-scale smoothed quantile regressions.

Theoretically, with p fixed and $n \to \infty$, Fernandes, Guerre and Horta (2019) showed that the convolution-type smoothed QR estimator has a lower asymptotic mean squared error than Horowitz's smoothed estimator, and also has a smaller Bahadur linearization error than the standard QR in the almost sure sense. The optimal order of the bandwidth based on the asymptotic mean squared error is unveiled as a function of n. Our theoretical results complement the results in Fernandes, Guerre and Horta (2019) by providing a comprehensive analysis of conquer in the regime in which p is allowed to increase with n.

2.3 Connection to related existing work

The idea of smoothing the piecewise linear loss, to the best of our knowledge, dates back to Amemiya (1982) in the context of median regression. Specifically, Amemiya (1982) proposed a smoothed approximation of the absolute value function that has the form

$$u \mapsto u + 2h \log(1 + e^{-u/h}), \quad u \in \mathbb{R},$$

where $h = n^{-\gamma}$ with $\gamma \in (1/3, 1/2)$. This smoothing device, however, is mainly used therein to simplify the analysis of the asymptotic behavior of a two-stage median regression estimator.

Horowitz (1998) proposed a kernel smoothing method of similar type for bootstrapping the median regression estimator. The idea is to replace the indicator function in $\rho_\tau(u) = u\{\tau - \mathbb{1}(u < 0)\}$

with a smoothed counterpart, leading to

$$\ell_h^{\text{Horo}}(u) = u\{\tau - \overline{K}(-u/h)\}, \tag{2.10}$$

where $\overline{K}(u) = \int_{-\infty}^{u} K(t)\,dt$, and $K(\cdot)$ is a symmetric kernel function. The key difference between the conquer loss (2.8) and Horowitz's loss (2.10) is that the former is globally convex, while Horowitz's loss is not. This is illustrated in Figure 2. We refer to Fernandes, Guerre and Horta (2019) for an in-depth comparison between convolution smoothing and Horowitz's smoothing in terms of asymptotic mean squared errors of the smoothed estimators.

A closer inspection reveals that the smoothed function considered by Amemiya (1982) is a convolution-type smoothed loss with a logistic kernel. More recently, Yi and Huang (2017) considered a smoothing approximation for the quantile loss, which they refer to as the *Huber approximation*. Their goal was to compute the regularized QR estimator but via a smoothed optimization method. The corresponding loss function also falls into the general framework of convolution-type smoothing, with K taken as the uniform kernel. The general statistical theory developed in this paper applies to those special cases.
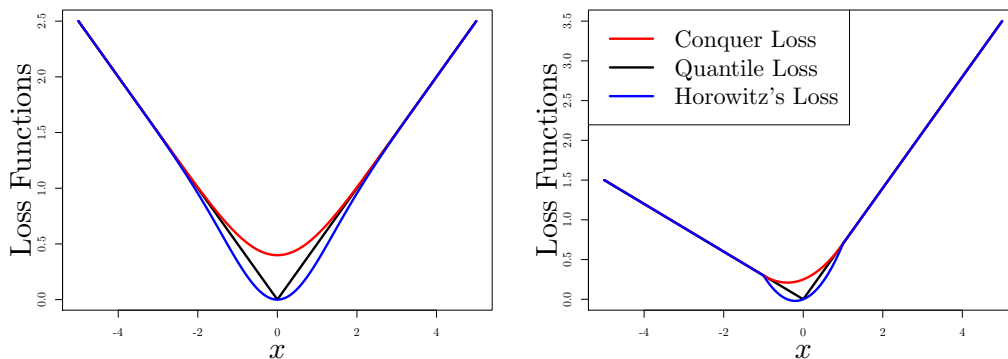
(a) Guassian kernel under $\tau = 0.5$. (b) Uniform kernel under $\tau = 0.7$.

Figure 2: Visualization of quantile loss (2.3), conquer loss (2.8), and Horowitz's smoothed loss (2.10) with Gaussian and uniform kernels, respectively.

2.4 Multiplier bootstrap inference

In this section, we consider a multiplier bootstrap procedure to construct confidence intervals for conquer. Independent of the observed sample $\mathcal{X}_n = \{(y_i, \boldsymbol{x}_i)\}_{i=1}^n$, let $\{w_i\}_{i=1}^n$ be independent and identically distributed random variables with $\mathbb{E}(w_i) = 1$ and $\text{var}(w_i) = 1$. Recall that $\widehat{\boldsymbol{\beta}}_h = \widehat{\boldsymbol{\beta}}_h(\tau) = \min_{\boldsymbol{\beta} \in \mathbb{R}^p} \widehat{Q}_h(\boldsymbol{\beta})$ is the conquer estimator. If the minimizer is not unique, we take any of the minima as $\widehat{\boldsymbol{\beta}}_h = (\widehat{\beta}_{h,1}, \ldots, \widehat{\beta}_{h,p})^{\mathrm{T}}$.

The proposed bootstrap method, which dates back to Dudewicz (1992) and Barbe and Bertail (1995), is based on reweighting the summands of $\widehat{Q}_h(\cdot)$ with random weights w_i. The resulting weighted quantile loss $\widehat{Q}_h^\flat : \mathbb{R}^p \to \mathbb{R}$ is

$$\widehat{Q}_h^\flat(\boldsymbol{\beta}) = \frac{1}{n} \sum_{i=1}^n w_i \ell_h(y_i - \langle \boldsymbol{x}_i, \boldsymbol{\beta} \rangle), \tag{2.11}$$

where $\ell_h(u) = (\rho_\tau * K_h)(u)$ is as in (2.8). We refer to Chatterjee and Bose (2005) for a general asymptotic theory for weighted bootstrap for estimating equations, where a class of bootstrap weights is considered. Extensions to semiparametric M-estimation can be found in Ma and Kosorok (2005) and Cheng and Huang (2010).

Let $\mathbb{E}^*$ and $\mathbb{P}^*$ be the conditional expectation and probability given the observed data $\mathcal{X}_n$, respectively. Observe that $\mathbb{E}^*\{\widehat{Q}_h^\flat(\boldsymbol{\beta})\} = \widehat{Q}_h(\boldsymbol{\beta})$ for any $\boldsymbol{\beta} \in \mathbb{R}^p$. Consequently, we have

$$\operatorname*{argmin}_{\boldsymbol{\beta} \in \mathbb{R}^p} \mathbb{E}^*\{\widehat{Q}_h^\flat(\boldsymbol{\beta})\} = \operatorname*{argmin}_{\boldsymbol{\beta} \in \mathbb{R}^p} \widehat{Q}_h(\boldsymbol{\beta}) = \widehat{\boldsymbol{\beta}}_h.$$

This simple and yet important observation motivates the following multiplier bootstrap statistic:

$$\widehat{\boldsymbol{\beta}}_h^\flat = \widehat{\boldsymbol{\beta}}_h^\flat(\tau) \in \operatorname*{argmin}_{\boldsymbol{\beta} \in \mathbb{R}^p} \widehat{Q}_h^\flat(\boldsymbol{\beta}). \tag{2.12}$$

To retain convexity of the loss function, non-negative random weights are preferred, such as $w_i \sim \text{Exp}(1)$, i.e., exponential distribution with rate 1, and $w_i = 1 + e_i$, where e_i are independent Rademacher random variables. We can construct confidence intervals based on the bootstrap estimates using one of the three classical methods, the percentile method, the pivotal method, and the normal-based method. To be specific, for each $q \in (0, 1)$ and $1 \le j \le p$, define the (conditional) q-quantile of $\widehat{\beta}_{h,j}^\flat$—the jth coordinate of $\widehat{\boldsymbol{\beta}}_h^\flat \in \mathbb{R}^p$—given the observed data as $\text{c}_j^\flat(q) = \inf\{t \in \mathbb{R} : \mathbb{P}^*(\widehat{\beta}_{h,j}^\flat \le t) \ge q\}$.

Then, for a prespecified nominal level $\alpha \in (0, 1)$, the corresponding $1 - \alpha$ bootstrap percentile and pivotal confidence intervals (CIs) for β_j^* ($j = 1, \ldots, p$) are, respectively,

$$\left[c_j^\flat(\alpha/2),\ c_j^\flat(1 - \alpha/2) \right] \quad \text{and} \quad \left[2\widehat{\beta}_{h,j} - c_j^\flat(1 - \alpha/2),\ 2\widehat{\beta}_{h,j} - c_j^\flat(\alpha/2) \right].$$

Numerically, $c_j^\flat(q)$ ($q \in \{\alpha, 1-\alpha/2\}$) can be calculated with any specified precision by the simulation. In the R package `conquer`, the default number of bootstrap replications is set as $B = 1000$.

In the next section, we will present a finite-sample theoretical framework for convolution-type smoothed quantile regression, including the concentration inequality and nonasymptotic Bahadur representation for both the conquer estimator (2.7) and its bootstrap counterpart (2.12) using Rademacher multipliers. As a by-product, a Berry-Esseen-type inequality (see Theorem 3.3) states that, under certain constraints on the (growing) dimensionality and bandwidth, the distribution of any linear projection of $\widehat{\beta}_h$ converges to a normal distribution as the sample size increases to infinity. Informally, for any given deterministic vector $\boldsymbol{a} \in \mathbb{R}^p$, the scaled statistic $n^{1/2} \langle \boldsymbol{a}, \widehat{\beta}_h^* - \beta^* \rangle$ is asymptotically normally distributed with asymptotic variance $\sigma_\tau^2 := \tau(1 - \tau)\, \boldsymbol{a}^\mathsf{T} \mathbf{D}^{-1} \boldsymbol{\Sigma} \mathbf{D}^{-1} \boldsymbol{a}$. To make inference based on such asymptotic results, we need to consistently estimate the asymptotic variance. Fernandes, Guerre and Horta (2019) suggested the following estimators

$$\widehat{\mathbf{D}}_h := \nabla^2 \widehat{Q}_h(\widehat{\beta}_h) = \frac{1}{n} \sum_{i=1}^n K(\widehat{\varepsilon}_i/h) \cdot \boldsymbol{x} \boldsymbol{x}_i^\mathsf{T} \quad \text{and} \quad \widehat{\mathbf{V}}_h := \frac{1}{nh} \sum_{i=1}^n \{\overline{K}(-\widehat{\varepsilon}_i/h) - \tau\}^2 \boldsymbol{x} \boldsymbol{x}^\mathsf{T}$$

of $\mathbf{D}$ and $\tau(1 - \tau)\, \boldsymbol{\Sigma}$, respectively, where $\widehat{\varepsilon}_i = y_i - \langle \boldsymbol{x}_i, \widehat{\beta}_h \rangle$ are fitted residuals. The ensuing $1 - \alpha$ normal-based CIs are given by $\widehat{\beta}_{h,j} \pm \Phi^{-1}(1 - \alpha/2) \cdot n^{-1/2} (\widehat{\mathbf{D}}_h^{-1} \widehat{\mathbf{V}}_h \widehat{\mathbf{D}}_h^{-1})_{jj}$, $j = 1, \ldots, p$. However, the normal approximations to the CI suffer from the sensitivity to the smoothing needed to estimated the conditional densities, namely, the matrix $\mathbf{D} = \mathbb{E}\{f_{\varepsilon|\boldsymbol{x}}(0)\boldsymbol{x}\boldsymbol{x}^\mathsf{T}\}$. This is especially true when τ is in the upper or lower tail. See Section 5.3 for a numerical comparison between normal approximation and bootstrap calibration for confidence construction at high and low quantile levels. As we shall see, the normal-based CIs can be exceedingly wide and thus inaccurate under these situations.

3 Statistical Analysis

Under the linear quantile regression model in (2.1), we write, for convenience, the generic data vector $(y, \boldsymbol{x})$ in a linear model form:

$$y = \langle \boldsymbol{x}, \boldsymbol{\beta}^*(\tau) \rangle + \varepsilon(\tau), \tag{3.1}$$

where the random variable $\varepsilon(\tau)$ satisfies $\mathbb{P}\{\varepsilon(\tau) \leq 0 | \boldsymbol{x}\} = \tau$. Let $f_{\varepsilon|\boldsymbol{x}}(\cdot)$ be the conditional density function of the regression error $\varepsilon = \varepsilon(\tau)$ given $\boldsymbol{x}$. We first derive upper bounds for the smoothing bias under mild regularity conditions on the conditional density $f_{\varepsilon|\boldsymbol{x}}$ and the kernel function.

3.1 Smoothing bias

Condition 3.1 (Kernel function). Let $K(\cdot)$ be a symmetric and non-negative function that integrates to one, that is, $K(u) = K(-u)$, $K(u) \geq 0$ for all $u \in \mathbb{R}$ and $\int_{-\infty}^{\infty} K(u) \, du = 1$. Moreover, $\int_{-\infty}^{\infty} u^2 K(u) \, du < \infty$.

We will use the notation $\kappa_k = \int_{-\infty}^{\infty} |u|^k K(u) \, du$ for $k \geq 1$. Furthermore, we define the population smoothed loss function $Q_h(\boldsymbol{\beta}) = \mathbb{E}\{\widehat{Q}_h(\boldsymbol{\beta})\}$, $\boldsymbol{\beta} \in \mathbb{R}^p$ and the pseudo parameter

$$\boldsymbol{\beta}_h^*(\tau) \in \underset{\boldsymbol{\beta} \in \mathbb{R}^p}{\arg\min} \, Q_h(\boldsymbol{\beta}), \tag{3.2}$$

which is the population minimizer under the smoothed quantile loss. For simplicity, we write $\boldsymbol{\beta}^* = \boldsymbol{\beta}^*(\tau)$ and $\boldsymbol{\beta}_h^* = \boldsymbol{\beta}_h^*(\tau)$ hereinafter. In general, $\boldsymbol{\beta}_h^*$ differs from $\boldsymbol{\beta}^*$, and we refer to $\|\boldsymbol{\beta}_h^* - \boldsymbol{\beta}^*\|_2$ as the approximation error or smoothing bias.

Condition 3.2 (Conditional density). There exist $\bar{f} \geq \underline{f} > 0$ such that $\underline{f} \leq f_{\varepsilon|\boldsymbol{x}}(0) \leq \bar{f}$ almost surely (for all $\boldsymbol{x}$). Moreover, there exists a constant $l_0 > 0$ such that $|f_{\varepsilon|\boldsymbol{x}}(u) - f_{\varepsilon|\boldsymbol{x}}(0)| \leq l_0 |u|$ for all $u \in \mathbb{R}$ almost surely.

Condition 3.3 (Random design: moments). The (random) vector $\boldsymbol{x} \in \mathbb{R}^p$ of covariates satisfies $\mu_3 := \sup_{\boldsymbol{u} \in \mathbb{S}^{p-1}} \mathbb{E}(|\langle \boldsymbol{\Sigma}^{-1/2} \boldsymbol{x}, \boldsymbol{u} \rangle|^3) < \infty$, where $\boldsymbol{\Sigma} = \mathbb{E}(\boldsymbol{x} \boldsymbol{x}^{\mathsf{T}})$ is positive definite.

If $\boldsymbol{x}$ is Gaussian, then Condition 3.3 holds trivially. Heavier-tailed distributions of $\boldsymbol{x}$ is excluded here so that we can expect standard rates of convergence for the quantile regression estimates. The

following result characterizes the smoothing bias from a nonasymptotic viewpoint.

Proposition 3.1. Assume Conditions 3.1–3.3 hold. There exist constants $c_1, c_2 > 0$ depending only on $(\underline{f}, l_0, \mu_3, \kappa_1, \kappa_2)$ such that for any $0 < h \leq c_1$, $\boldsymbol{\beta}_h^*$ is the unique minimizer of $\boldsymbol{\beta} \mapsto Q_h(\boldsymbol{\beta})$ and satisfies

$$\|\boldsymbol{\beta}_h^* - \boldsymbol{\beta}^*\|_{\mathbf{D}} \leq c_2 h^2, \tag{3.3}$$

with $\mathbf{D} = \mathbb{E}\{f_{\varepsilon|\boldsymbol{x}}(0)\boldsymbol{x}\boldsymbol{x}^{\mathsf{T}}\}$. In addition, assume $\kappa_3 < \infty$, $f_{\varepsilon|\boldsymbol{x}}(\cdot)$ is continuously differentiable and satisfies almost surely that $|f'_{\varepsilon|\boldsymbol{x}}(u) - f'_{\varepsilon|\boldsymbol{x}}(0)| \leq l_1|u|$ for some constant $l_1 > 0$. Then

$$\left\|\boldsymbol{\beta}_h^* - \boldsymbol{\beta}^* + \frac{1}{2}\kappa_2 h^2 \cdot \mathbf{D}^{-1}\mathbb{E}\{f'_{\varepsilon|\boldsymbol{x}}(0)\boldsymbol{x}\}\right\|_{\mathbf{D}} \leq c_3 h^3, \tag{3.4}$$

where $c_3 > 0$ is a constant depending only on $(\underline{f}, l_0, l_1, \mu_3)$ and the kernel K.

To better understand inequality (3.3), consider for simplicity a special case where $f_{\varepsilon|\boldsymbol{x}}(0)$ is a constant (independent of $\boldsymbol{x}$). Hence, $\mathbf{D} = f_{\varepsilon|\boldsymbol{x}}(0)\boldsymbol{\Sigma}$ and (3.3) becomes $(\mathbb{E}\langle\boldsymbol{x}, \boldsymbol{\beta}_h^* - \boldsymbol{\beta}^*\rangle^2)^{1/2} = \|\boldsymbol{\beta}_h^* - \boldsymbol{\beta}^*\|_{\boldsymbol{\Sigma}} \leq f_{\varepsilon|\boldsymbol{x}}^{-1/2}(0)c_2 h^2$. On the left-hand side, $\|\boldsymbol{\beta}_h^* - \boldsymbol{\beta}^*\|_{\boldsymbol{\Sigma}}$ is the average prediction smoothing error. Interestingly, the upper bound on the right-hand side could be dimension-free given h. Another interesting implication is that, when both $f_{\varepsilon|\boldsymbol{x}}(0)$ and $f'_{\varepsilon|\boldsymbol{x}}(0)$ are independent of $\boldsymbol{x}$, i.e., $f_{\varepsilon|\boldsymbol{x}}(0) = f_\varepsilon(0)$ and $f'_{\varepsilon|\boldsymbol{x}}(0) = f'_\varepsilon(0)$, the leading term in the bias simplifies to

$$\frac{1}{2}\kappa_2 h^2 \cdot \mathbf{D}^{-1}\mathbb{E}\{f'_{\varepsilon|\boldsymbol{x}}(0)\boldsymbol{x}\} = \frac{f'_\varepsilon(0)}{2f_\varepsilon(0)}\kappa_2 h^2 \cdot \boldsymbol{\Sigma}^{-1}\mathbb{E}(\boldsymbol{x}) = \frac{f'_\varepsilon(0)}{2f_\varepsilon(0)}\kappa_2 h^2 \cdot \begin{bmatrix} 1 \\ \mathbf{0}_{p-1} \end{bmatrix}.$$

In other words, the smoothing bias is concentrated primarily on the intercept. In the asymptotic setting where p is fixed, and $h = o(1)$ as $n \to \infty$, we refer to Theorem 1 in Fernandes, Guerre and Horta (2019) for the expression of asymptotic bias.

3.2 Finite sample theory

In this section, we provide two nonasymptotic results, the concentration inequality and the Bahadur-Kiefer representation, for the conquer estimator under random design.

Condition 3.4 (Random design: sub-exponential case). The predictor $x \in \mathbb{R}^p$ is *sub-exponential*: there exists $v_0 > 0$ such that $\mathbb{P}\{|\langle u, w \rangle| \geq v_0 \|u\|_2 t\} \leq 2e^{-t}$ for all $u \in \mathbb{R}^p$ and $t \geq 0$, where $w = \Sigma^{-1/2}x$ with $\Sigma = \mathbb{E}(xx^{\mathsf{T}})$ positive-definite.

Condition 3.4 assumes a sub-exponential condition on the random covariates, which encompasses the bounded case considered by Fernandes, Guerre and Horta (2019). For the standardized predictor $w = \Sigma^{-1/2}x$, we define the uniform moment parameters

$$\mu_k = \sup_{u \in \mathbb{S}^{p-1}} \mathbb{E}|\langle u, w \rangle|^k, \quad k = 1, 2, \ldots, \tag{3.5}$$

with $\mu_2 = 1$. Under Condition 3.4, a straightforward calculation shows that $\mu_k \leq 2v_0^k k!$, valid for all $k \geq 1$, where v_0 is defined in Condition 3.4. In general, v_0 may increase with p, but in this paper we are primarily interested in the cases where v_0 is a dimension-free constant. This is indeed the case if x is multivariate Gaussian, multivariate Bernoulli, uniform on $[-1, 1]^p$, or uniform on the sphere/ball with radius $p^{1/2}$; see Section 3.4 in Vershynin (2018) for more examples.

Theorem 3.1. Assume that Conditions 3.1–3.4 hold with $\kappa_l = \min_{|u| \leq 1} K(u) > 0$. For any $t > 0$, the smoothed quantile regression estimator $\widehat{\beta}_h$ with $\sqrt{(p + t)/n} \lesssim h \lesssim 1$ satisfies

$$\|\widehat{\beta}_h - \beta^*\|_{\Sigma} \leq C(\kappa_l \underline{f})^{-1} \left(\sqrt{\frac{p + t}{n}} + h^2 \right), \tag{3.6}$$

with probability at least $1 - 2e^{-t}$, where $C > 0$ is a constant depending only on (v_0, κ_2, l_0).

The estimation error in (3.6) is upper bounded by two terms, h^2 and $\sqrt{(p + t)/n}$, and can be interpreted as the upper bounds for bias and variance, respectively. The condition $\min_{|u| \leq 1} K(u) > 0$ can be relaxed to $\min_{|u| \leq c} K(u) > 0$ for some $c \in (0, 1)$, which will only change the constants encountered in the proof. In particular, for kernels that are compactly supported on $[-1, 1]$, we may choose $c = 1/2$ in Theorem 3.1. Next, we establish a Bahadur representation for conquer. To this end, we impose a slightly more stringent sub-Gaussian condition on the covariates.

Condition 3.5 (Random design: sub-Gaussian case). The predictor $x \in \mathbb{R}^p$ is *sub-Gaussian*: there exists $v_1 > 0$ such that $\mathbb{P}\{|\langle u, w \rangle| \geq v_1 \|u\|_2 t\} \leq 2e^{-t^2/2}$ for all $u \in \mathbb{R}^p$ and $t \geq 0$, where $w = \Sigma^{-1/2}x$.

We are primarily concerned with the cases where v_1 is a dimension-free constant that does not

depend on p; see the remarks made earlier about υ_0. Assume that the data $\{(y_i, \boldsymbol{x}_i)\}_{i=1}^n$ are generated from the conditional quantile model (2.1), and write $\varepsilon_i = y_i - \langle \boldsymbol{x}_i, \boldsymbol{\beta}^* \rangle$ which satisfy $\mathbb{P}(\varepsilon_i \leq 0 | \boldsymbol{x}_i) = \tau$.

Theorem 3.2. Assume Conditions 3.1, 3.2, and 3.5 hold with $\kappa_l = \min_{|u| \leq 1} K(u) > 0$ and $\kappa_u = \sup_{u \in \mathbb{R}} K(u)$. Moreover, assume $\sup_{u \in \mathbb{R}} f_{\varepsilon|\boldsymbol{x}}(u) \leq \bar{f}$ almost surely. For any $t > 0$, let the bandwidth h satisfy $\sqrt{(p+t)/n} \lesssim h \lesssim 1$. Then,

$$\left\| \widehat{\boldsymbol{\beta}}_h - \boldsymbol{\beta}^* - \mathbf{D}^{-1} \frac{1}{n} \sum_{i=1}^n \{\tau - \bar{K}(-\varepsilon_i/h)\} \boldsymbol{x}_i \right\|_2 \leq C\left(\frac{p+t}{h^{1/2}n} + h\sqrt{\frac{p+t}{n}} + h^3 \right), \qquad (3.7)$$

with probability at least $1 - 3e^{-t}$, where $\mathbf{D} = \mathbb{E}\{f_{\varepsilon|\boldsymbol{x}}(0)\boldsymbol{x}\boldsymbol{x}^{\mathrm{T}}\}$, and $C > 0$ is a constant depending only on $(\upsilon_1, \kappa_2, \kappa_u, \kappa_l, l_0, \bar{f}, \underline{f})$.

The Bahadur representation can be used to establish the limiting distribution of the estimator or its functionals. Here we consider a fundamental statistical inference problem for testing the linear hypothesis $H_0 : \langle \boldsymbol{a}, \boldsymbol{\beta}^* \rangle = 0$, where $\boldsymbol{a} \in \mathbb{R}^p$ is a deterministic vector that defines a linear functional of interest. It is then natural to consider a test statistic that depends on $n^{1/2}\langle \boldsymbol{a}, \widehat{\boldsymbol{\beta}}_h \rangle$. Based on the nonasymptotic result in Theorem 3.2, we establish a Berry-Esseen bound for the linear projection of conquer.

Theorem 3.3. Assume that the conditions in Theorem 3.2 hold, and $\sqrt{(p + \log n)/n} \lesssim h \lesssim 1$. Then,

$$\Delta_{n,p}(h) := \sup_{x \in \mathbb{R}, \, \boldsymbol{a} \in \mathbb{R}^p} \left| \mathbb{P}\left\{ \frac{n^{1/2}\langle \boldsymbol{a}, \widehat{\boldsymbol{\beta}}_h - \boldsymbol{\beta}^* \rangle}{\sqrt{\tau(1-\tau)\,\boldsymbol{a}^{\mathrm{T}}\mathbf{D}^{-1}\boldsymbol{\Sigma}\mathbf{D}^{-1}\boldsymbol{a}}} \leq x \right\} - \Phi(x) \right| \lesssim \frac{p + \log n}{(nh)^{1/2}} + n^{1/2}h^2, \qquad (3.8)$$

where $\Phi(\cdot)$ denotes the standard normal distribution function. In particular, with the choice of bandwidth $h \asymp \{(p + \log n)/n\}^{2/5}$, the normal approximation error $\Delta_{n,p}(h)$ is of order $(p + \log n)^{4/5} n^{-3/10}$.

It is worth noticing that although the bound in (3.8) holds uniformly over all $\boldsymbol{a}$, the asymptotic variance $\sigma_\tau^2 := \tau(1-\tau)\,\boldsymbol{a}^{\mathrm{T}}\mathbf{D}^{-1}\boldsymbol{\Sigma}\mathbf{D}^{-1}\boldsymbol{a}$ depends on $\boldsymbol{a}$.

Remark 3.1 (Large-p asymptotics)**.** A broader view of classical asymptotics recognizes that the parametric dimension of appropriate model sequences may tend to infinity with the sample size; that is $p = p_n \to \infty$ as $n \to \infty$. Results with increasing p are available in the context of regularized quantile regression (Belloni and Chernozhukov, 2011; Wang, Wu and Li, 2012; Koenker et al., 2017). In the large-p and larger-n setting with quantile regression estimation without regularization, Welsh (1989) shows that $p^3(\log n)^2/n \to 0$ suffices for a normal approximation, which

provides some support to the viability of observed rates of parametric growth in the applied literature. Koenker (1988) considers a sample of 733 wage models appeared in the econometric literature, and finds that $p_n = O(n^{1/4})$ is roughly consistent with empirical practice.

For the (convolution-type) smoothed quantile regression, the Berry-Esseen bound (3.8) in Theorem 3.3 immediately yields a large-p asymptotic result. Taking $h = h_n = \{(p + \log n)/n\}^{2/5}$ therein, we find that $n^{1/2}\langle a, \widehat{\beta}_h - \beta^* \rangle$, for any given $a \in \mathbb{R}^p$, is asymptotically normally distributed as long as $p^{8/3}/n \to 0$, which improves the best known growth condition on p for quantile regression (Welsh, 1989). For smooth robust regression estimators, asymptotic normality can be proven under less restrictive conditions on p. Huber (1973) showed that if the loss is twice differentiable, the asymptotic normality for $\langle a, \widehat{\beta} \rangle$, where $a \in \mathbb{R}^p$, holds if $p^3/n \to 0$ as n increases. Portnoy (1985) weakened the condition to $(p \log n)^{3/2}/n \to 0$ if the loss function is four times differentiable and the error distribution is symmetric. For Huber loss that has a Lipschitz continuous derivative, He and Shao (2000) obtained the scaling $p^2 \log p = o(n)$ that ensures the asymptotic normality of arbitrary linear combinations of $\widehat{\beta}$. Table 1 summarizes our discussion here and show that the smoothing for conquer helps ensure asymptotic normality of the estimator under weaker conditions on p than what we need for the usual quantile regression estimator.

Table 1: Summary of scaling conditions required for normal approximation under various loss functions.

Loss function	Design	Scaling condition
Huber loss (Huber, 1973)	Fixed design	$p^3 = o(n)$
Four times differentiable loss (Portnoy, 1985)	Fixed design (with symmetric error)	$(p \log n)^{3/2} = o(n)$
Huber loss (He and Shao, 2000)	Fixed design	$p^2 \log p = o(n)$
Huber loss (Chen and Zhou, 2020)	Sub-Gaussian	$p^2 = o(n)$
Quantile loss (Welsh, 1989; He and Shao, 2000)	Fixed design	$p^3 (\log n)^2 = o(n)$
Quantile loss (Pan and Zhou, 2020)	Sub-Gaussian	$p^3 (\log n)^2 = o(n)$
Convolution smoothed quantile loss	Sub-Gaussian	$p^{8/3} = o(n)$

Remark 3.2. In this paper, we show that the accuracy of conquer-based inference via the Bahadur representation (and normal approximations) has an error of rate faster than $n^{-1/4}$ yet slower than $n^{-1/2}$; see Theorems 3.2 and 3.3. For standard regression quantiles, Portnoy (2012) proposed an alternative expansion for the quantile process using the "Hungarian" construction of Komlós, Major and Tusnády. This stochastic approximation yields an error of order $n^{-1/2}$ (up to a factor of $\log n$), and hence provides a theoretical justification for accurate approximations for inference in regression quantile models.

3.3 Theoretical guarantees for bootstrap inference

We next investigate the statistical properties of the bootstrapped estimator defined in (2.12), with a particular focus on the Rademacher multiplier bootstrap. To be specific, we use, in this section and the rest of the paper, the random weights $w_i = 1 + e_i$ for $i = 1, \ldots, n$, where $e_1, \ldots, e_n$ are independent Rademacher random variables, that is, $\mathbb{P}(e_i = 1) = \mathbb{P}(e_i = -1) = 1/2$. As before, we consider array (non)asymptotics, so that the bootstrap approximation errors depend explicitly on (n, p).

Theorem 3.4. Assume Conditions 3.1–3.3, and 3.5 hold with $\kappa_l = \min_{|u| \le 1} K(u) > 0$. For any $t \ge 0$, there exists some event $\mathcal{E}(t)$ with $\mathbb{P}\{\mathcal{E}(t)\} \ge 1 - 5e^{-t}$ such that, with $\mathbb{P}^*$-probability at least $1 - 2e^{-t}$ conditioned on $\mathcal{E}(t)$, the bootstrap estimator $\widehat{\beta}_h^{\flat}$ given in (2.12) satisfies

$$\|\widehat{\beta}_h^{\flat} - \beta^*\|_{\Sigma} \le C\left(\sqrt{\frac{p + t}{n}} + h^2 \right) \tag{3.9}$$

as long as $\max\{(p + \log n + t)/n, \sqrt{(p + t)/n}\} \lesssim h \lesssim 1$, where $C > 0$ is a constant depending only on $(\upsilon_1, \kappa_2, \kappa_l, l_0, \underline{f})$.

Analogously to Theorem 3.2, we further provide a Bahadur representation result for the bootstrap counterpart $\widehat{\beta}_h^{\flat}$, which lays the theoretical foundation for the validity of Rademacher multiplier bootstrap for conquer.

Theorem 3.5. Assume Conditions 3.1–3.3 and 3.5 hold with $\kappa_l = \min_{|u| \le 1} K(u) > 0$ and $\kappa_u = \sup_{u \in \mathbb{R}} K(u)$. Moreover, assume $\sup_{u \in \mathbb{R}} f_{\varepsilon|x}(u) \le \bar{f}$ almost surely in x. Let the bandwidth satisfy $h \asymp \{(p + \log n)/n\}^{\delta}$ for some $\delta \in [1/3, 1/2)$. Provided that $n \gtrsim p + \log n$, there exists a sequence of events $\{\mathcal{F}_n\}$ with $\mathbb{P}(\mathcal{F}_n) \ge 1 - 6n^{-1}$ such that, with $\mathbb{P}^*$-probability at least $1 - 2n^{-1}$ conditioned on $\mathcal{F}_n$,

$$\widehat{\beta}_h^{\flat} - \widehat{\beta}_h = \mathbf{D}^{-1} \frac{1}{n} \sum_{i=1}^{n} e_i \cdot x_i \{\tau - \overline{K}(-\varepsilon_i/h)\} + r_n^{\flat}, \tag{3.10}$$

where $r_n^{\flat}$ depends on $\{(e_i, y_i, x_i)\}_1^n$ and satisfies $\|r_n^{\flat}\|_2 = O_{\mathbb{P}^*}(\chi_n)$, where $\chi_n = \chi_n(\{y_i, x_i\}_{i=1}^n) = O_{\mathbb{P}}\{\sqrt{(p + \log n)p \log n}/(h^{1/2}n) + (p + \log n)^{1/2}p \log(n)/n\}$.

As suggested by Theorem 3.3 and Remark 3.1, if we take h to be $\{(p + \log n)/n\}^{2/5}$, the normal approximation to the conquer estimator is asymptotically accurate provided that $p^{8/3} = o(n)$ as $n \to$

18

∞. For the same h, the remainder $r_n^\flat$ in (3.10) satisfies $\|r_n^\flat\|_2 = o_{\mathbb{P}^*}(n^{-1/2})$ as long as $p^3(\log n)^2 = o(n)$. Taken these two parts together, we have the following asymptotic bootstrap approximation result. If the dimension $p = p_n$ satisfies $p^3(\log n)^2 = o(n)$, then, as $n \to \infty$,

$$\sup_{x \in \mathbb{R}} \left| \mathbb{P}(n^{1/2} \langle a, \widehat{\beta}_h - \beta^* \rangle \le x) - \mathbb{P}^*(n^{1/2} \langle a, \widehat{\beta}_h^\flat - \widehat{\beta}_h \rangle \le x) \right| \xrightarrow{\mathbb{P}} 0. \tag{3.11}$$

The proof of (3.11) follows the same argument as that in the proof of Theorem 3.3, and therefore is omitted. The additional logarithmic factor in the scaling may be an artifact of the proof technique. For standard quantile regression, Feng, He and Hu (2011) established a fixed-p asymptotic bootstrap approximation result for wild bootstrap under fixed design.

4   Gradient Descent Methods for Conquer

To solve optimization problems (2.7) and (2.12) with non-negative weights, arguably the simplest algorithm is a vanilla gradient descent algorithm (GD). For a prespecified $\tau \in (0, 1)$ and bandwidth $h > 0$, recall that $\widehat{Q}_h(\beta) = (1/n) \sum_{i=1}^n \ell_h(y_i - \langle x_i, \beta \rangle)$. Starting with an initial value $\widehat{\beta}^0 \in \mathbb{R}^p$, at iteration $t = 0, 1, 2, \ldots$, GD computes

$$\widehat{\beta}^{t+1} = \widehat{\beta}^t - \eta_t \cdot \nabla \widehat{Q}_h(\widehat{\beta}^t), \tag{4.1}$$

where $\eta_t > 0$ is the stepsize and

$$\nabla \widehat{Q}_h(\beta) = \frac{1}{n} \sum_{i=1}^n \left\{ \overline{K}\left(-\frac{y_i - \langle x_i, \beta \rangle}{h} \right) - \tau \right\} x_i \quad \text{with} \quad \overline{K}(u) = \int_{-\infty}^u K(t) \, dt.$$

In the classical GD method, the stepsize is usually obtained by employing line search techniques. However, line search is computationally intensive for large-scale settings. One of the most important issues in GD is to determine a proper update step η_t decay schedule. A common practice in the literature is to use a diminishing stepsize or a best-tuned fixed stepsize. Neither of these two approaches can be efficient, at least compared to the Newton-Frisch algorithm with preprocessing (Portnoy and

Koenker, 1997). Recall that the smoothed loss $\widehat{Q}_h(\cdot)$ is twice differentiable with Hessian

$$\nabla^2 \widehat{Q}_h(\boldsymbol{\beta}) = \frac{1}{nh} \sum_{i=1}^n K\left(\frac{y_i - \langle \boldsymbol{x}_i, \boldsymbol{\beta} \rangle}{h}\right) \boldsymbol{x}_i \boldsymbol{x}_i^\mathsf{T}.$$

It is therefore intriguing to employ the Newton-Raphson method, which at iteration t would read

$$\widehat{\boldsymbol{\beta}}^{t+1} = \widehat{\boldsymbol{\beta}}^t - \{\nabla^2 \widehat{Q}_h(\widehat{\boldsymbol{\beta}}^t)\}^{-1} \nabla \widehat{Q}_h(\widehat{\boldsymbol{\beta}}^t). \tag{4.2}$$

Computing the inverse of the Hessian for large problems can be an expensive operation. Moreover, when h is small, $\nabla^2 \widehat{Q}_h(\cdot)$ can have a very large condition number, which leads to slow convergence.

Remark 4.1. Below we list the explicit expressions of the convolution smoothed check functions using several commonly used kernels. Recall that the check function can be written as $\rho_\tau(u) = |u|/2 + (\tau - 1/2)u$, which, after convolution smoothing, becomes

$$\ell_h(u) = \int_{-\infty}^{\infty} \frac{1}{2}|u + hv| K(v) \, \mathrm{d}v + (\tau - 1/2)u.$$

- (Uniform kernel $K(u) = (1/2)\mathbb{1}(|u| \le 1)$): $\ell_h(u) = (h/2)\ell^\mathrm{U}(u/h) + (\tau - 1/2)u$, where $\ell^\mathrm{U}(u) := (u^2/2 + 1/2)\mathbb{1}(|u| \le 1) + |u|\mathbb{1}(|u| > 1)$ is a Huber-type loss (Huber, 1973).

- (Gaussian kernel $K(u) = (2\pi)^{-1/2}e^{-u^2/2}$): $\ell_h(u) = (h/2)\ell^\mathrm{G}(u/h) + (\tau - 1/2)u$, where $\ell^\mathrm{G}(u) := (2/\pi)^{1/2}e^{-u^2/2} + u\{1 - 2\Phi(-u)\}$.

- (Logistic kernel $K(u) = e^{-u}/(1 + e^{-u})^2$): $\ell_h(u) = (h/2)\ell^\mathrm{L}(u/h) + (\tau - 1/2)u$, where $\ell^\mathrm{L}(u) := u + 2\log(1 + e^{-u})$.

- (Epanechnikov kernel $K(u) = (3/4)(1 - u^2)\mathbb{1}(|u| \le 1)$): $\ell_h(u) = (h/2)\ell^\mathrm{E}(u/h) + (\tau - 1/2)u$, where $\ell^\mathrm{E}(u) := (3u^2/4 - u^4/8 + 3/8)\mathbb{1}(|u| \le 1) + |u|\mathbb{1}(|u| > 1)$.

- (Triangular kernel $K(u) = (1 - |u|)\mathbb{1}(|u| \le 1)$): $\ell_h(u) = (h/2)\ell^\mathrm{T}(u/h) + (\tau - 1/2)u$, where $\ell^\mathrm{T}(u) := (u^2 - |u|^3/3 + 1/3)\mathbb{1}(|u| \le 1) + |u|\mathbb{1}(|u| > 1)$.

4.1 The Barzilai-Borwein stepsize rule

In this paper, we propose to solve conquer by means of the gradient descent with a Barzilai-Borwein update step (Barzilai and Borwein, 1988), which we refer to as the GD-BB algorithm. The BB

method, which is motivated by quasi-Newton methods, has been proven to be very successful in solving nonlinear optimization problems.

Recall the Newton-Raphson method (4.2), which can be computationally expensive and needs modifications if the Hessian is ill-conditioned especially when h is too small. For this reason, many quasi-Newton methods seek a simple approximation of the inverse Hessian matrix, say $(\mathbf{H}^t)^{-1}$, which satisfies the secant equation $\mathbf{H}^t \boldsymbol{\delta}^t = \boldsymbol{g}^t$, where for $t = 1, 2, \ldots$,

$$\boldsymbol{\delta}^t = \widehat{\boldsymbol{\beta}}^t - \widehat{\boldsymbol{\beta}}^{t-1} \quad \text{and} \quad \boldsymbol{g}^t = \nabla \widehat{Q}_h(\widehat{\boldsymbol{\beta}}^t) - \nabla \widehat{Q}_h(\widehat{\boldsymbol{\beta}}^{t-1}). \tag{4.3}$$

To relieve the computational bottleneck of inverting a large matrix, the BB method chooses η so that $\eta \nabla \widehat{Q}_h(\widehat{\boldsymbol{\beta}}^t) = (\eta^{-1} \mathbf{I}_p)^{-1} \nabla \widehat{Q}_h(\widehat{\boldsymbol{\beta}}^t)$ "approximates" $(\mathbf{H}^t)^{-1} \nabla \widehat{Q}_h(\widehat{\boldsymbol{\beta}}^t)$. Since $\mathbf{H}^t$ satisfies $\mathbf{H}^t \boldsymbol{\delta}^t = \boldsymbol{g}^t$, it is more practical to choose η such that $(1/\eta)\boldsymbol{\delta}^t \approx \boldsymbol{g}^t$ or $\boldsymbol{\delta}^t \approx \eta \boldsymbol{g}^t$. Via least squares approximations, one may use $\eta_{1,t}^{-1} = \operatorname{argmin}_\alpha \|\alpha \boldsymbol{\delta}^t - \boldsymbol{g}^t\|_2^2$ or $\eta_{2,t} = \operatorname{argmin}_\eta \|\boldsymbol{\delta}^t - \eta \boldsymbol{g}^t\|_2^2$. The BB stepsizes are then defined as

$$\eta_{1,t} = \frac{\langle \boldsymbol{\delta}^t, \boldsymbol{\delta}^t \rangle}{\langle \boldsymbol{\delta}^t, \boldsymbol{g}^t \rangle} \quad \text{and} \quad \eta_{2,t} = \frac{\langle \boldsymbol{\delta}^t, \boldsymbol{g}^t \rangle}{\langle \boldsymbol{g}^t, \boldsymbol{g}^t \rangle}. \tag{4.4}$$

Consequently, the BB iteration takes the form

$$\widehat{\boldsymbol{\beta}}^{t+1} = \widehat{\boldsymbol{\beta}}^t - \eta_{\ell,t} \nabla \widehat{Q}_h(\widehat{\boldsymbol{\beta}}^t), \quad \ell = 1 \text{ or } 2. \tag{4.5}$$

Note that the BB step starts at iteration 1, while at iteration 0, we compute $\widehat{\boldsymbol{\beta}}^1$ using the standard gradient descent with an initial estimate $\widehat{\boldsymbol{\beta}}^0$. The procedure is summarized in Algorithm 1. Based on extensive numerical studies, we find that at a fixed τ, the number of iterations is insensitive to varying (n, p) combinations. Moreover, as h increases, the number of iterations declines because the loss function is "more convex" for larger h. In Algorithm 1, the quantity $\delta > 0$ is called the gradient tolerance, ensuring that the obtained estimate, say $\widehat{\boldsymbol{\beta}}$, satisfies $\|\nabla \widehat{Q}_h(\widehat{\boldsymbol{\beta}})\|_2 \leq \delta$. Provided that $\delta \lesssim \sqrt{p/n}$, the statistical theory developed in Section 3 prevails. In our R package `conquer`, we set $\delta = 10^{-4}$ as the default value; this value can also be specified by the user.

As τ approaches 0 or 1, the Hessian matrix becomes ill-conditioned. As a result, the stepsizes computed in GD-BB may sometimes vibrate drastically, causing instability of the algorithm.

Algorithm 1 Gradient descent with Barzilai-Borwin stepsize (GD-BB) for solving conquer.

Input: data vectors $\{(y_i, \boldsymbol{x}_i)\}_{i=1}^n$, $\tau \in (0,1)$, bandwidth $h \in (0,1)$, initialization $\widehat{\boldsymbol{\beta}}^{(0)}$, and gradient tolerance δ.

1:   Compute $\widehat{\boldsymbol{\beta}}^1 \leftarrow \widehat{\boldsymbol{\beta}}^0 - \nabla\widehat{Q}_h(\widehat{\boldsymbol{\beta}}^0)$
2:   **for** $t = 1, 2 \ldots$ **do**
3:      $\boldsymbol{\delta}^t \leftarrow \widehat{\boldsymbol{\beta}}^t - \widehat{\boldsymbol{\beta}}^{t-1}$, $\boldsymbol{g}^t \leftarrow \nabla\widehat{Q}_h(\widehat{\boldsymbol{\beta}}^t) - \nabla\widehat{Q}_h(\widehat{\boldsymbol{\beta}}^{t-1})$
4:      $\eta_{1,t} \leftarrow \langle\boldsymbol{\delta}^t, \boldsymbol{\delta}^t\rangle / \langle\boldsymbol{\delta}^t, \boldsymbol{g}^t\rangle$, $\eta_{2,t} \leftarrow \langle\boldsymbol{\delta}^t, \boldsymbol{g}^t\rangle / \langle\boldsymbol{g}^t, \boldsymbol{g}^t\rangle$
5:      $\eta_t \leftarrow \min\{\eta_{1,t}, \eta_{2,t}, 100\}$ if $\eta_{1,t} > 0$ and $\eta_t \leftarrow 1$ otherwise
6:      $\widehat{\boldsymbol{\beta}}^{t+1} \leftarrow \widehat{\boldsymbol{\beta}}^t - \eta_t\nabla\widehat{Q}_h(\widehat{\boldsymbol{\beta}}^t)$
7:   **end for** when $\|\nabla\widehat{Q}_h(\widehat{\boldsymbol{\beta}}^t)\|_2 \leq \delta$

Therefore, in practice, we set a lower bound for the stepsizes by taking $\eta_t = \min\{\eta_{1,t}, \eta_{2,t}, 100\}$, for $t = 1, 2, \ldots$. Another cause of an ill-conditioned Hessian arises when we have covariates with very different scales. In this case, the stepsize should be different for each covariate, and a constant stepsize will be either too small or too large for one or more covariates, which correspond to slow convergence. To address this issue, we scale the covariate inputs to have zero mean and unit variance before applying gradient descent.

4.2 Warm start via Huber regression

A good initialization helps reduce the number of iterations for GD, and hence facilitates fast convergence. Recall from Remark 4.1 that with a uniform kernel, the smoothed check function is proximal to a Huber loss (Huber, 1973). Motivated by this subtle proximity, we propose using the Huber M-estimator as an initial estimate, and then proceed by iteratively applying gradient descent with BB update step.

Let $H_\gamma(u) = (u^2/2)\mathbb{1}(|u| \leq \gamma) + \gamma(|u| - \gamma/2)\mathbb{1}(|u| > \gamma)$ be the Huber loss parametrized by $\gamma > 0$. The Huber M-estimator is then defined as

$$\widetilde{\boldsymbol{\beta}}_\gamma \in \underset{\boldsymbol{\beta}\in\mathbb{R}^p}{\operatorname{argmin}} \widehat{\mathcal{H}}_\gamma(\boldsymbol{\beta}), \quad \text{where} \quad \widehat{\mathcal{H}}_\gamma(\boldsymbol{\beta}) = \frac{1}{n}\sum_{i=1}^n H_\gamma(y_i - \langle\boldsymbol{x}_i, \boldsymbol{\beta}\rangle). \tag{4.6}$$

Note that $H_1(u) = \ell^{\mathrm{U}}(u) - 1/2$ for $\ell^{\mathrm{U}}(\cdot)$ defined in Remark 4.1, we have $\widehat{Q}_{1/2,\gamma}(\boldsymbol{\beta}) = \widehat{\mathcal{H}}_\gamma(\boldsymbol{\beta})$. The quantity γ is a shape parameter that controls the amount of robustness. The main reason for choosing a fixed (neither diminishing nor diverging) tuning parameter γ in Huber (1981) is to guarantee robustness towards arbitrary contamination in a neighborhood of the model. This is at the core

---

Algorithm 2 GD-BB method for solving (4.6).

Input: $\{(y_i, \boldsymbol{x}_i)\}_{i=1}^n$ and convergence criterion δ.

1: Initialize $\widetilde{\boldsymbol{\beta}}^{(0)} = \boldsymbol{0}$
2: Compute $\gamma^0 = 1.35 \cdot \text{MAD}(\{r_i^0\}_{i=1}^n)$, where $r_i^0 \leftarrow y_i - \langle \boldsymbol{x}_i, \widetilde{\boldsymbol{\beta}}^0 \rangle$, $i = 1, \ldots, n$, where $\text{MAD}(\cdot)$ is the median absolute deviation
3: $\widetilde{\boldsymbol{\beta}}^1 \leftarrow \widetilde{\boldsymbol{\beta}}^0 - \nabla \widehat{L}_{\gamma^0}(\widetilde{\boldsymbol{\beta}}^0)$
4: **for** $t = 1, 2 \ldots$ **do**
5: $\quad \gamma^t = 1.35 \cdot \text{MAD}(\{r_i^t\}_{i=1}^n)$, where $r_i^t \leftarrow y_i - \langle \boldsymbol{x}_i, \widetilde{\boldsymbol{\beta}}^t \rangle$, $i = 1, \ldots, n$
6: $\quad \boldsymbol{\delta}^t \leftarrow \widetilde{\boldsymbol{\beta}}^t - \widetilde{\boldsymbol{\beta}}^{t-1}$, $\boldsymbol{g}^t \leftarrow \nabla \widehat{\mathcal{H}}_{\gamma^t}(\widetilde{\boldsymbol{\beta}}^t) - \nabla \widehat{\mathcal{H}}_{\gamma^t}(\widetilde{\boldsymbol{\beta}}^{t-1})$
7: $\quad \eta_{1,t} \leftarrow \langle \boldsymbol{\delta}^t, \boldsymbol{\delta}^t \rangle / \langle \boldsymbol{\delta}^t, \boldsymbol{g}^t \rangle$, $\eta_{2,t} \leftarrow \langle \boldsymbol{\delta}^t, \boldsymbol{g}^t \rangle / \langle \boldsymbol{g}^t, \boldsymbol{g}^t \rangle$.
8: $\quad \eta_t \leftarrow \min\{\eta_{1,t}, \eta_{2,t}, 100\}$ if $\eta_{1,t} > 0$ and $\eta_t \leftarrow 1$ otherwise
9: $\quad \widetilde{\boldsymbol{\beta}}^{t+1} \leftarrow \widetilde{\boldsymbol{\beta}}^t - \eta_t \nabla \widehat{\mathcal{H}}_{\gamma^t}(\widetilde{\boldsymbol{\beta}}^t)$
10: **end for** when $\|\nabla \widehat{\mathcal{H}}_{\gamma^t}(\widetilde{\boldsymbol{\beta}}^t)\|_2 \leq \delta$

---

of the robust statistics idiosyncrasy. In particular, Huber (1981) proposed $\gamma = 1.35\sigma$ to gain as much robustness as possible while retaining 95% asymptotic efficiency for normally distributed data, where $\sigma > 0$ is the standard deviation of the random noise. We estimate σ using the median absolute deviation on the residuals at each iteration, i.e., $\text{MAD}(\{r_i^t\}_{i=1}^n) = \text{median}(|r_i^t - r_{\text{median}}^t|)$, where r_{median}^t is the median of $r_1^t, \ldots, r_n^t$.

Noting that the Huber loss is continuously differentiable, convex, and locally strongly convex, we use the GD-BB method described in the previous section to solve the optimization problem (4.6). Starting at iteration 0 with $\widetilde{\boldsymbol{\beta}}^0 = \boldsymbol{0}$, at iteration $t = 0, 1, 2, \ldots$, we compute

$$\widetilde{\boldsymbol{\beta}}^{t+1} = \widetilde{\boldsymbol{\beta}}^t - \eta_t \nabla \widehat{\mathcal{H}}_\gamma(\widetilde{\boldsymbol{\beta}}^t) = \widetilde{\boldsymbol{\beta}}^t + \frac{\eta_t}{n} \sum_{i=1}^n \psi_\gamma(y_i - \langle \boldsymbol{x}_i, \widetilde{\boldsymbol{\beta}}^t \rangle) \boldsymbol{x}_i \tag{4.7}$$

with $\eta_t > 0$ automatically obtained by the BB method, where $\psi_\gamma(u) = H_\gamma'(u) = \min\{\max(-\gamma, u), \gamma\}$. We summarize the details in Algorithm 2.

Remark 4.2. The Huber loss $H_\gamma(\cdot)$ approximates the quantile loss function with $\tau = 1/2$ as $\gamma \to 0$. Therefore, an alternative method for QR computing is to solve the Huber regression via gradient descent with a shrinking gamma. To evaluate its performance, we implement the above idea by setting $\gamma^t = c \cdot \gamma^{t-1}$ for some $c \in (0, 1)$ at the t-th iteration. We found that the aforementioned idea is not stable numerically across several simulated data sets, unless one is very careful in controlling the minimal magnitude of γ. In addition, the solution obtained has a higher estimation error than that of standard QR and conquer.

5 Numerical Studies

In this section, we assess the finite-sample performance of conquer via extensive numerical studies. We compare conquer to standard QR (Koenker and Bassett, 1978) and Horowitz's smoothed QR (Horowitz, 1998). Both the convolution-type and Horowitz's smoothed methods involve a smoothing parameter h. In view of Theorem 3.3, we take $h = \{(p + \log n)/n\}^{2/5}$ in all of the numerical experiments. As we will see from Figure 5, the proposed method is insensitive to the choice of h. Therefore, we leave the fine tuning of h as an optional rather than imperative choice. In all the numerical experiments, the convergence criterion in Algorithms 1 and 2 is taken as $\delta = 10^{-4}$.

We first generate the covariates $\boldsymbol{x}_i = (x_{i,1}, \ldots, x_{i,p})^{\mathrm{T}}$ from a multivariate uniform distribution on the cube $3^{1/2} \cdot [-1, 1]^p$ with covariance matrix $\boldsymbol{\Sigma} = (0.7^{|j-k|})_{1 \le j,k \le p}$ using the R package MultiRNG (Falk, 1999). The random noise ε_i are generated from two different distributions: (i) Gaussian distribution, $\mathcal{N}(0, 4)$; and (ii) t distribution with degrees of freedom 2, t_2. Let $\boldsymbol{x}_i^- = (x_{i,1}, \ldots, x_{i,p-1})^{\mathrm{T}}$, $\boldsymbol{\beta}^* = (1, \ldots, 1)^{\mathrm{T}}_{p-1}$, and $\beta_0^* = 1$. Given $\tau \in (0, 1)$, we then generate the response y_i from the following homogeneous and heterogeneous models, all of which satisfy the Assumption 2.1:

1. Homogeneous model:

$$y_i = \beta_0^* + \langle \boldsymbol{x}_i^-, \boldsymbol{\beta}^* \rangle + \{\varepsilon_i - F_{\varepsilon_i}^{-1}(\tau)\}, \quad i = 1, \ldots, n; \tag{5.1}$$

2. Linear heterogeneous model:

$$y_i = \beta_0^* + \langle \boldsymbol{x}_i^-, \boldsymbol{\beta}^* \rangle + (0.5x_{i,p} + 1)\{\varepsilon_i - F_{\varepsilon_i}^{-1}(\tau)\}, \quad i = 1, \ldots, n; \tag{5.2}$$

3. Quadratic heterogeneous model:

$$y_i = \beta_0^* + \langle \boldsymbol{x}_i^-, \boldsymbol{\beta}^* \rangle + 0.5\{1 + (x_{i,p} - 1)^2\}\{\varepsilon_i - F_{\varepsilon_i}^{-1}(\tau)\}, \quad i = 1, \ldots, n. \tag{5.3}$$

To evaluate the performance of different methods, we calculate the estimation error under the ℓ_2-norm, i.e., $\|\widehat{\boldsymbol{\beta}} - \boldsymbol{\beta}^*\|_2$, and record the elapsed time. The details are in Section 5.1. In Section 5.2, we examine the finite-sample performance of the multiplier bootstrap method for constructing confidence intervals in terms of coverage probability, width of the interval, and computing time.
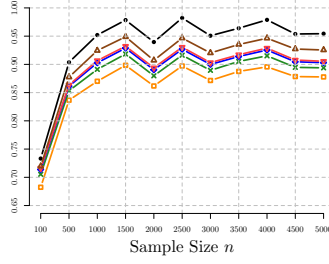
5.1 Estimation

For all the numerical studies in this section, we consider a wide range of the sample size n, with the size-dimension ratio fixed at $n/p = 20$. That is, we allow the dimension p to increase as a function of n. We implement conquer with four different kernel functions as described in Remark 4.1: (i) Gaussian; (ii) uniform; (iii) Epanechnikov; and (iv) triangular. The classical quantile regression is implemented via a modified version of the Barrodale and Roberts algorithm (Koenker and d'Orey, 1987, 1994) by setting `method=` "`br`" in the R package `quantreg`, which is recommended for problems up to several thousands of observations in Koenker (2019). For very large problems, the Frisch-Newton approach after preprocessing "pfn" is preferred. Since the same size taken to be at most 5000 throughout this section, the two methods, "br" and "pfn", have nearly identical runtime behaviors. In some applications where there are a lot of discrete covariates, it is advantageous to use method "sfn", a sparse version of Frisch-Newton algorithm that exploits sparse algebra to compute iterates (Koenker and Ng, 2003). Moreover, we implement Horowitz's smoothed quantile regression using the Gaussian kernel, and solve the resulting non-convex optimization via gradient descent with random initialization and stepsize calibrated by backtracking line search (Section 9.3 of Boyd and Vandenberghe, 2004). The results, averaged over 100 replications, are reported.

Figure 3 depicts estimation error of the different methods under the simulation settings described in Section 5 with $\tau = 0.9$. We see that conquer has a lower estimation error than the classical QR across all scenarios, indicating that smoothing can improve estimation accuracy under the finite-sample setting. Moreover, compared to Horowitz's smoothing, conquer has a lower estimation error in most settings. Estimation error under various quantile levels $\tau \in \{0.1, 0.3, 0.5, 0.7\}$ under the t_2 random noise are also examined. The results are reported in Figure D.1 in Appendix D, from which we observe evident advantages of conquer, especially at low and high quantile levels.

To assess the computational efficiency, we compute the elapsed time for fitting the different methods. Figures 4 and D.2 in Appendix D report the runtime for the different methods with growing sample size and dimension under the same settings as in Figures 3 and D.1, respectively. We observe that conquer is computationally efficient and stable across all scenarios, and the runtime is insensitive to the choice of kernel functions. In contrast, the runtime for classical quantile regression grows rapidly as the sample size and dimension increase. Figures 4 and D.2 in Appendix D show that the runtime of Horowitz's smoothing method increases significantly at extreme quantile
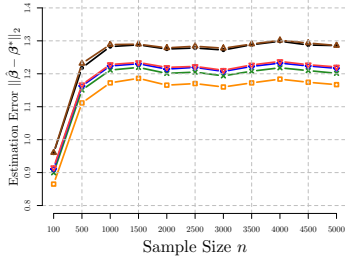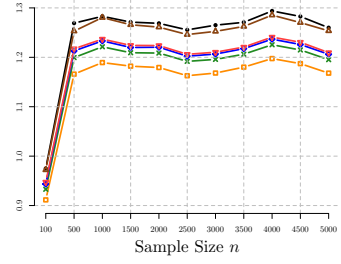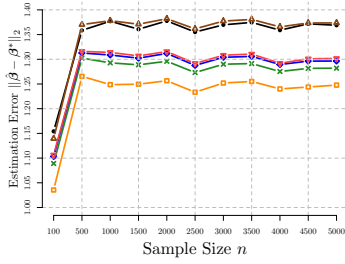
25

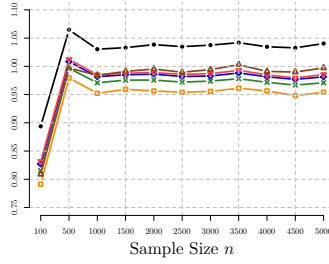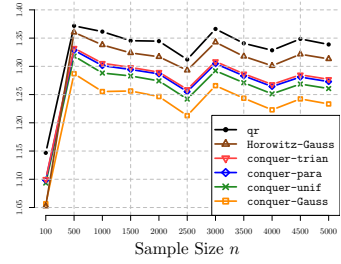|		
(a) Model (5.1) with $\mathcal{N}(0,4)$ error.	(b) Model (5.2) with $\mathcal{N}(0,4)$ error.	(c) Model (5.3) with $\mathcal{N}(0,4)$ error.
(d) Model (5.1) with t_2 error.	(e) Model (5.2) with t_2 error.	(f) Model (5.3) with t_2 error.

Figure 3: Estimation error under models (5.1)–(5.3) in Section 5 with $\mathcal{N}(0,4)$ and t_2 errors, $\tau = 0.9$, averaged over 100 data sets for three different methods: (i) quantile regression `qr`, (ii) Horowitz's method with Gaussian kernel `Horowitz-Gauss`, and (iii) the conquer method with four different kernel functions `conquer-trian`, `conquer-para`, `conquer-unif`, and `conquer-Gauss`.

levels $\tau \in \{0.1, 0.9\}$, possibly due to the combination of its non-convex nature and flatter gradient. In summary, we conclude that conquer significantly improves computational efficiency while retaining high statistical accuracy for fitting large-scale linear quantile regression models.

Next, we conduct a sensitivity analysis for conquer regarding the smoothing bandwidth h. We set $(n, p) = (2000, 100)$ and consider the simulation settings (5.1)–(5.3) with $\mathcal{N}(0, 4)$ and t_2 noise. We perform conquer with $h \in \{0.2, 0.22, \ldots, 0.5\}$, including the default value $h_{\mathrm{de}} = \{(p + \log n)/n\}^{2/5} = 0.3107$, and compare the estimation error with that of QR in Figure 5. We see that the estimation error of conquer is uniformly lower than that of QR over a range of h, suggesting that conquer is insensitive to the choice of bandwidth h.

5.2 Inference

In this section, we assess the performance of the multiplier bootstrap procedure for constructing confidence interval for each of the regression coefficients obtained from conquer. We implement
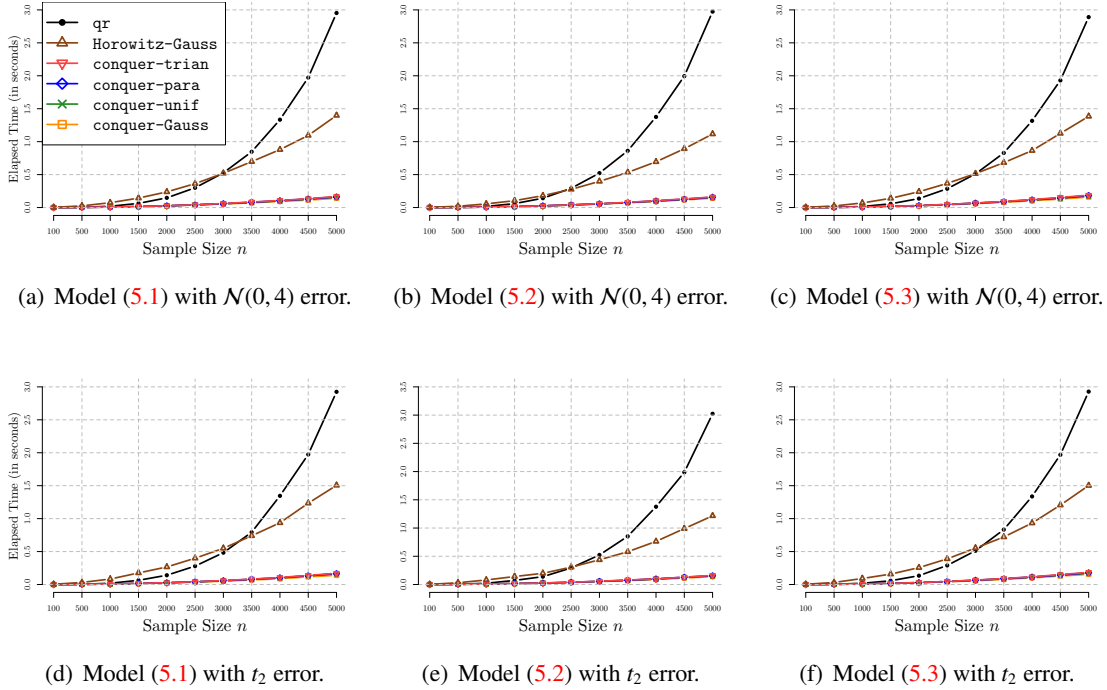
(a) Model (5.1) with $\mathcal{N}(0,4)$ error.　(b) Model (5.2) with $\mathcal{N}(0,4)$ error.　(c) Model (5.3) with $\mathcal{N}(0,4)$ error.

(d) Model (5.1) with t_2 error.　(e) Model (5.2) with t_2 error.　(f) Model (5.3) with t_2 error.

Figure 4: Elapsed time of standard QR, Horowitz's smoothing, and conquer when $\tau = 0.9$. The model settings are the same as those in Figure 3.

conquer using the Gaussian kernel, and construct three types of confidence intervals: (i) the percentile `mb-per`; (ii) pivotal `mb-piv`; (iii) and regular `mb-norm` confidence intervals, as described in Section 2.4. We also refer to the proposed multiplier bootstrap procedure as `mb-conquer` for simplicity. We compare the proposed method to several widely used inference methods for QR. In particular, we consider confidence intervals by inverting a rank score test, `rank` (Gutenbrunner and Jurečková (1992); Section 3.5 of Koenker (2005)); a bootstrap variant based on pivotal estimating functions, `pwy` (Parzen, Wei and Ying, 1994); and wild bootstrap with Rademacher weights, `wild` (Feng, He and Hu, 2011). The three methods `rank`, `pwy`, and `wild` are implemented using the R package `quantreg`. Note that `rank` is a non-resampling based procedure that relies on prior knowledge on the random noise, i.e., a user needs to specify whether the random noise are independent and identically distributed. In our simulation studies, we provide `rank` an unfair advantage by specifying the correct random noise structure.

We set $(n, p) = (800, 20)$, $\tau \in \{0.5, 0.9\}$, and significance level $\alpha = 0.05$. All of the resampling methods are implemented with $B = 1000$ bootstrap samples. To measure the reliability, accuracy,
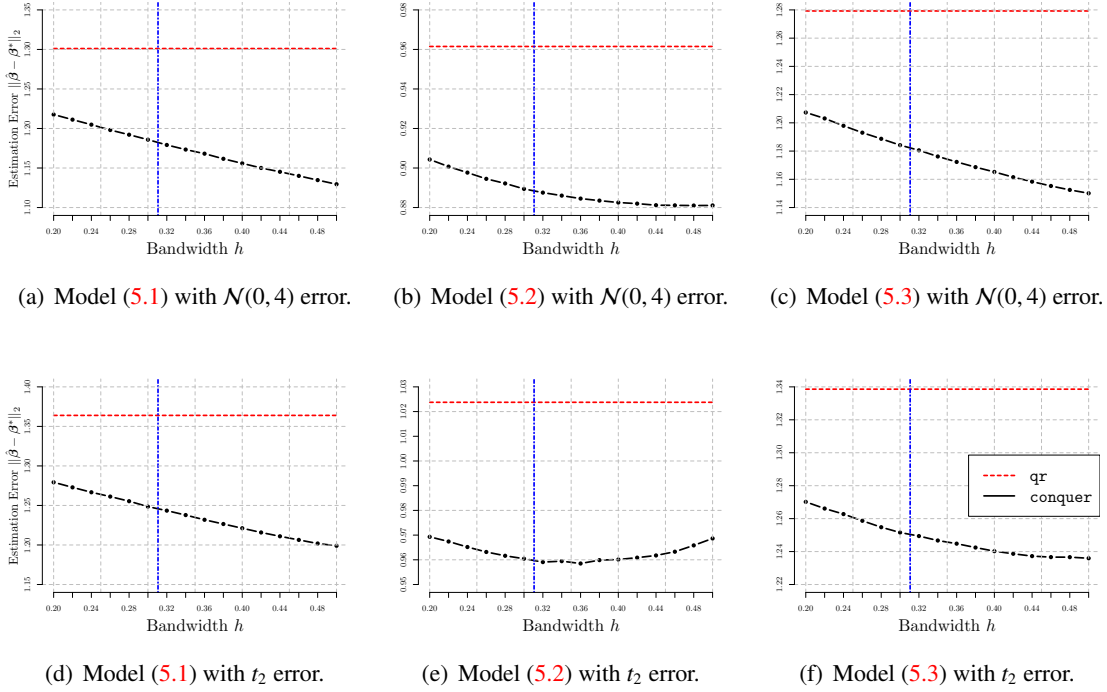
(a) Model (5.1) with $\mathcal{N}(0, 4)$ error.   (b) Model (5.2) with $\mathcal{N}(0, 4)$ error.   (c) Model (5.3) with $\mathcal{N}(0, 4)$ error.

(d) Model (5.1) with t_2 error.   (e) Model (5.2) with t_2 error.   (f) Model (5.3) with t_2 error.

Figure 5: Sensitivity analysis of conquer with a range of bandwidth parameter h. Results for $(n, p) = (2000, 100)$, averaged over 100 datasets, with conquer implemented using a Gaussian kernel. The blue vertical dash line represents our default choice $h_{\mathrm{de}} = \{(p + \log n)/n\}^{2/5}$, and the red horizontal dash line represents the estimation error of standard QR.

and computational efficiency of different methods for constructing confidence intervals, we calculate the average empirical coverage probability, average width of confidence interval, and the average runtime. The average is taken over all regression coefficients without the intercept. Results based on 200 replications are reported in Figure 6, and Figures D.3–D.5 in Appendix D.

In Figure 6, and Figures D.3–D.5 in Appendix D, we use the rank-inversion method, rank, as a benchmark since we implement rank using information about the true underlying random noise, which is practically infeasible. In the case of $\tau = 0.9$, pwy is most conservative as it produces the widest confidence intervals with slightly inflated coverage probability, and wild gives the narrowest confidence intervals but at the cost of coverage probability. The proposed methods mb-per, mb-piv, and mb-norm achieve a good balance between reliability (high coverage probability) and accuracy (narrow CI width), and moreover, has the lowest runtime.

To further highlight the computational gain of the proposed method, we now perform numerical studies with larger n and p. In this case, the rank inversion method rank is computationally infeasi-
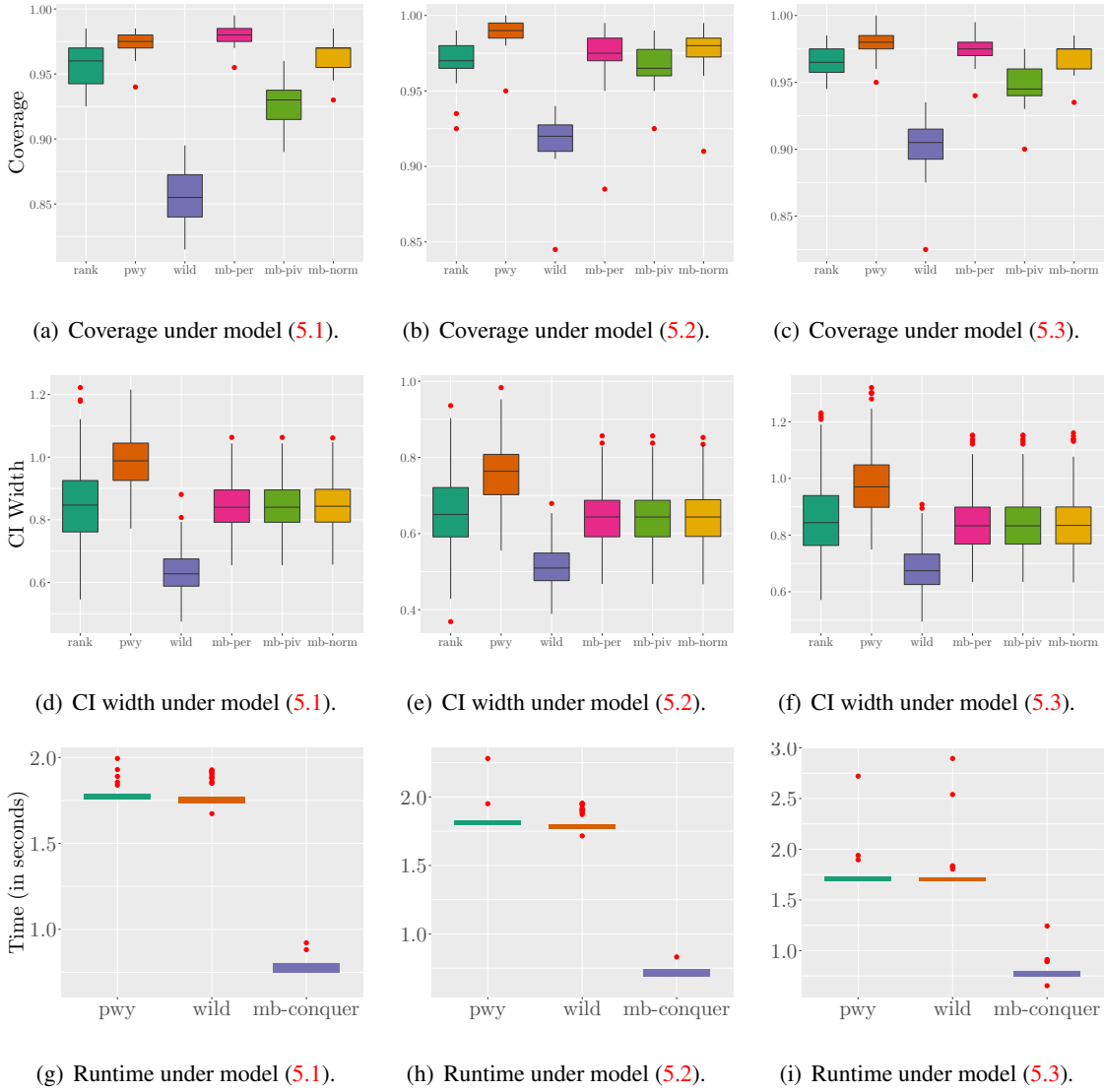
(a) Coverage under model (5.1). (b) Coverage under model (5.2). (c) Coverage under model (5.3).

(d) CI width under model (5.1). (e) CI width under model (5.2). (f) CI width under model (5.3).

(g) Runtime under model (5.1). (h) Runtime under model (5.2). (i) Runtime under model (5.3).

Figure 6: Empirical coverage, confidence interval width, and elapsed time of *six* methods: `rank`, `pwy`, `wild` and three types of mb-conquer: `mb-per`, `mb-piv`, and `mb-norm` under models (5.1)–(5.3) with t_2 errors. For the running time, `rank` is not included since it is not a resampling-based method. The quantile level τ is fixed to be 0.9, and the results are averaged over 200 data sets.

ble. For example, when $(n, p) = (5000, 250)$, rank inversion takes approximately 80 minutes while conquer with multiplier bootstrap takes 41 seconds for constructing confidence intervals. We therefore omit `rank` from the following comparison. We consider the quadratic heterogeneous model (5.3) with $(n, p) = (4000, 100)$ and t_2 noise. The results are reported in Figure 7. We see that `pwy` and `wild` take approximately 300 seconds while `mb-conquer` takes approximately 15 seconds. In summary, `mb-conquer` leads to a huge computational gain without sacrificing statistical efficiency.

29

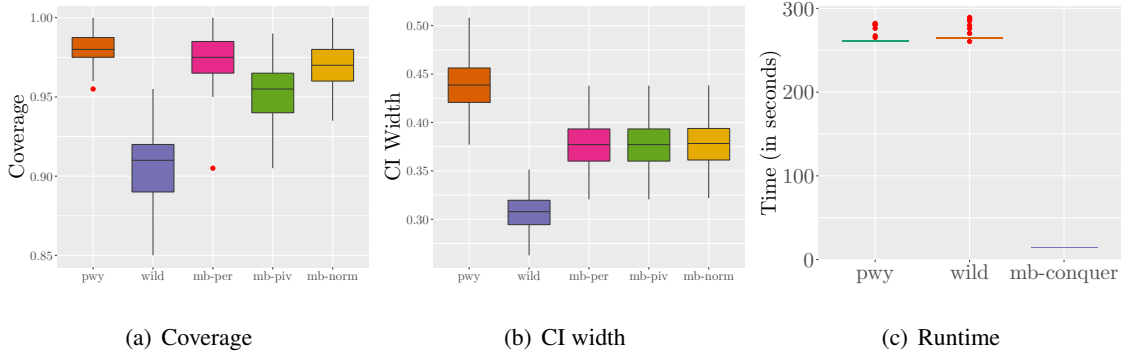| (a) Coverage	(b) CI width	(c) Runtime

Figure 7: Empirical coverage, confidence interval width and elapsed time of `pwy`, `wild` and 3 types of mb-conquer: `mb-per`, `mb-piv`, and `mb-norm` under quadratic heterogeneous model (5.3) with t_2 errors. This figure extends the rightmost column of Figure 6 to larger scale: $(n, p) = (4000, 100)$.

5.3 Comparison between normal approximation and bootstrap calibration

Finally, we complement the above studies with a comparison between the normal approximation and bootstrap calibration methods for confidence estimation. We consider model (5.1) with $(n, p) = (2000, 10)$. We use the same $\beta^* \in \mathbb{R}^{p-1}$ and β_0^* as before, and generate random covariates and noise from a multivariate uniform distribution and $t_{1.5}$-distribution, respectively. For each of the $p - 1$ regression coefficients, we apply the proposed bootstrap percentile method and the normal-based method (Fernandes, Guerre and Horta, 2019) to construct pointwise confidence intervals at quantile indices close to 0 and 1, that is, $\tau \in \{0.05, 0.1, 0.9, 0.95\}$.

Boxplots of the empirical coverage and CI width for the two methods are reported in Figure 8. Considering that extreme quantile regressions are notoriously hard to estimate, the bootstrap method can produce much more reliable (high coverage) and accurate (narrow width) confidence intervals than the normal-based counterpart. Therefore, for applications in which extreme quantiles are of particular interest, such as the problem of forecasting the conditional value-at-risk of a financial institution (Chernozhukov and Umantsev, 2001), the bootstrap provides a more reliable approach for quantifying the uncertainty of the estimates.

6 Discussion

In this paper, we provide a comprehensive study on the statistical properties of `conquer`—namely, convolution-type smoothed quantile regression, under the array nonasymptotic setting in which p is

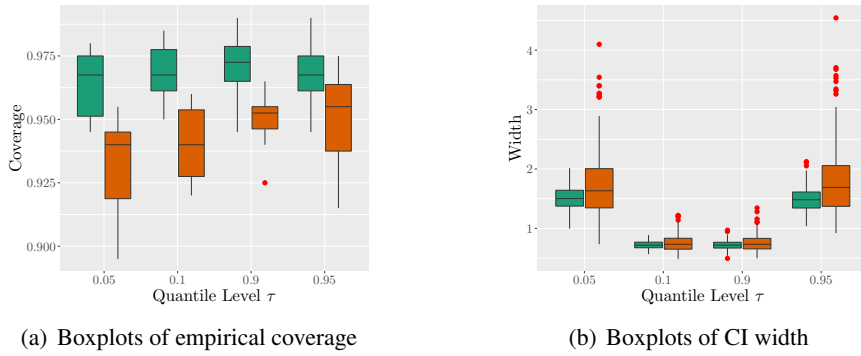| (a) Boxplots of empirical coverage | (b) Boxplots of CI width |

Figure 8: Boxplots of the empirical coverage and CI width for the bootstrap percentile method ▉ and normal-based method ▉ under model (5.1) with $t_{1.5}$ error.

allowed to increase as a function of n while p/n being small. An efficient gradient-based algorithm is proposed to compute conquer, which proves to be scalable to large dimensions and even larger sample sizes. Recently, there has been a growing interest in studying the asymptotic behavior of regression estimates under the regime in which p grows proportionally with the sample size n (Donoho and Montanari, 2016; Lei, Bickel and El Karoui, 2018). The current studies are mainly focused on M-estimation with strongly convex loss functions. New theoretical tools are needed to establish results for conquer under such a regime, and we leave it for future work.

In the high-dimensional setting in which $p \gg n$, various authors have studied the regularized quantile regression under the sparsity assumption that most of the regression coefficients are zero (Belloni and Chernozhukov, 2011; Wang, Wu and Li, 2012). Using ℓ_1 penalties, the computation of regularized QR is based on either a linear programming reformulation or alternating direction method of multiplier algorithms (Gu et al., 2018). Since the conquer loss is convex and twice differentiable, we expect that gradient-based algorithms, such as coordinate gradient descent or proximal gradient descent, will enjoy superior computational efficiency for solving regularized conquer without sacrificing statistical accuracy. One future work is to establish the statistical theory and computational complexity of the regularized conquer estimator.

References

AMEMIYA, T. (1982). Two stage least absolute deviations estimators. *Econometrica* **50** 689–711.

ARCONES, M. A. (1996). The Bahadur-Kiefer representation of L_p regression estimators. *Econom.*

Theory **12** 257–283.

BARBE, P. and BERTAIL, P. (1995). *The Weighted Bootstrap. Lecture Notes in Statistics* **98**. Springer, New York.

BARRODALE, I. and ROBERTS, F. (1974). Solution of an overdetermined system of equations in the ℓ_1 norm. *Communications of the ACM* **17** 319–320.

BARZILAI, J. and BORWEIN, J. M. (1988). Two-point step size gradient methods. *IMA J. Numer. Anal.* **8** 141–148.

BELLONI, A. and CHERNOZHUKOV, V. (2011). ℓ_1-regularized quantile regression in high-dimensional sparse models. *Ann. Statist.* **39** 82–130.

BICKEL, P. J. (1975). One-step Huber estimates in the linear model. *J. Amer. Statist. Assoc.* **70** 428–434.

BOYD, S. and VANDENBERGHE, L. (2004). *Convex Optimization.* Cambridge University Press, Cambridge.

CHATTERJEE, S. and BOSE, A. (2005). Generalized bootstrap for estimating equations. *Ann. Statist.* **33** 414–436.

CHEN, X., LIU, W. and ZHANG, Y. (2019). Quantile regression under memory constraint. *Ann. Statist.* **47** 3244–3273.

CHEN, X. and ZHOU, W.-X. (2020). Robust inference via multiplier bootstrap. *Ann. Statist.* **48** 1665–1691.

CHENG, G. and HUANG, J. Z. (2010). Bootstrap consistency for general semiparametric *M*-estimation. *Ann. Statist.* **38** 2884–2915.

CHERNOZHUKOV, V. and UMANTSEV, L. (2001). Conditional value-at-risk: Aspects of modeling and estimation. *Empirical Economics* **26** 271–293.

DE CASTRO, L., GALVAO, A. F., KAPLAN, D. M. and LIU, X. (2019). Smoothed GMM for quantile models. *J. Econom.* **213** 121–144.

DONOHO, D. and MONTANARI, A (2016). High dimensional robust *M*-estimation: asymptotic variance via approximate message passing. *Probab. Theory Related Fields* **166** 935–969.

DUDEWICZ, E. J. (1992). The Generalized Bootstrap. In *Bootstrapping and Related Techniques. Lecture Notes in Economics and Mathematical Systems* **376** 31–37. Springer-Verlag, Berlin.

EDDELBUETTEL, D. and SANDERSON, C. (2014). RcppArmadillo: Accelerating R with high-performance C++ linear algebra. *Comput. Statist. Data Anal.* **71** 1054–1063.

FALK, M. (1999). A simple approach to the generation of uniformly distributed random variables with prescribed correlations. *Comm. Statist. Simulation Comput.* **28** 785–791.

FENG, X., HE, X. and HU, J. (2011). Wild bootstrap for quantile regression. *Biometrika* **98** 995–999.

FERNANDES, M., GUERRE, E. and HORTA, E. (2019). Smoothing quantile regressions. *J. Bus. Econ. Statist.*, in press.

GALVAO, A. F. and KATO, K. (2016). Smoothed quantile regression for panel data. *J. Econom.* **193** 92–112.

GU, Y., FAN, J., KONG, L., MA, S. and ZOU, H. (2018). ADMM for high-dimensional sparse regularized quantile regression. *Technometrics* **60** 319–331.

GUTENBRUNNER, C. and JUREČKOVÁ, J. (1992). Regression rank scores and regression quantiles. *Ann. Statist.* **20** 305–330.

HE, X. and SHAO, Q.-M. (1996). A general Bahadur representation of *M*-estimators and its application to linear regression with nonstochastic designs. *Ann. Statist.* **24** 2608–2630.

HE, X. and HU, F. (2002). Markov chain marginal bootstrap. *J. Amer. Statist. Assoc.* **97** 783–795.

HE, X. and SHAO, Q.-M. (2000). On parameters of increasing dimensions. *J. Mult. Anal.* **73** 120–135.

HOROWITZ, J. L. (1998). Bootstrap methods for median regression models. *Econometrica* **66** 1327–1351.

HUBER, P. J. (1973). Robust estimation: Asymptotics, conjectures and Monte Carlo. *Ann. Statist.* **1** 799–821.

HUBER, P. J. (1981). *Robust Statistics*. John Wiley & Sons, New York.

KAPLAN, D. M. and SUN, Y. (2017). Smoothed estimating equations for instrumental variables quantile regression. *Econom. Theory* **33** 105–157.

KOCHERGINSKY, M., HE, X. and HU, F. (2005). Practical confidence intervals for regression quantiles. *J. Comp. Graph. Statist.* **14** 41–55.

KOENKER, R. (1988). Asymptotic theory and econometric practice. *J. Appl. Econom.* **3** 139–147.

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

KOENKER, R. (2019). Package "quantreg", version 5.54. Reference manual: https://cran.r-project.org/web/packages/quantreg/quantreg.pdf.

KOENKER, R. and BASSETT, G. (1978). Regression quantiles. *Econometrica* **46** 33–50.

KOENKER, R., CHERNOZHUKOV, V., HE, X. and PENG, L. (2017). *Handbook of Quantile Regression.* CRC Press, New York.

KOENKER, R. and D'OREY, V. (1987). Computing quantile regressions. *J. R. Statist. Soc. C* **36** 383–393.

KOENKER, R. and D'OREY, V. (1994). A remark on Algorithm AS 229: Computing dual regression quantiles and regression rank scores. *J. R. Statist. Soc. C* **43** 410–414.

KOENKER, R. and NG, P. (2003). SparseM: A sparse matrix package for R. *J. Statist. Software* **8** 1–9.

LEI, L., BICKEL, P. J. and EL KAROUI, N. (2018). Asymptotic for high dimensional regression *M*-estimates: fixed design results. *Probab. Theory Related Fields* **172** 983–1079.

MA, S. and KOSOROK, M. R. (2005). Robust semiparametric *M*-estimation and the weighted bootstrap. *J. Mult. Anal.* **96** 190–217.

NEMIROVSKI, A. and YUDIN, D. (1983). *Problem Complexity and Method Efficiency in Optimization.* Wiley, 1983.

PAN, X. and ZHOU, W.-X. (2020). Multiplier bootstrap for quantile regression: Non-asymptotic theory under random design. *Information and Inference: A Journal of the IMA*, to appear.

PARIKH, N. and BOYD, S. (2014). Proximal algorithms. *Found. Trends Optim.* **1** 127–239.

PARZEN, M. I., WEI, L. J. and YING, Z. (1994). A resampling method based on pivotal estimating functions. *Biometrika* **81** 341–350.

PORTNOY, S. (1985). Asymptotic behavior of *M*-estimators of p regression parameters when p^2/n is large; II. Normal approximation. *Ann. Statist.* **13** 1403–1417.

PORTNOY, S. (2012). Nearly root-n approximation for regression quantile processes. *Ann. Statist.* **40** 1714–1736.

PORTNOY, S. and KOENKER, R. (1997). The Gaussian hare and the Laplacian tortoise: Computability of squared-error versus absolute-error estimators. *Statist. Sci.* **12** 279–300.

POWELL, J. L. (1991). Estimation of monotonic regression models under quantile restrictions. In *Nonparametric and Semiparametric Methods in Econometrics* (eds W. Barnett, J. Powell and G. Tauchen). Cambridge Univ. Press, Cambridge.

VAN DER VAART, A. W. and WELLNER, J. A. (1996). *Weak Convergence and Empirical Processes: With Applications to Statistics.* Springer, New York.

VERSHYNIN, R. (2018). *High-Dimensional Probability: An Introduction with Applications in Data Science.* Cambridge University Press, Cambridge.

WANG, H. J., STEFANSKI, L. A. and ZHU, Z. (2012). Corrected-loss estimation for quantile regression with covariate measurement errors. *Biometrika* **99** 405–421.

WANG, L., WU, Y. and LI, R. (2012). Quantile regression for analyzing heterogeneity in ultra-high dimension. *J. Amer. Statist. Assoc.* **107** 214–222.

WELSH, A. H. (1989). On *M*-processes and *M*-estimation. *Ann. Statist.* **15** 337–361.

WHANG, Y.-J. (2006). Smoothed empirical likelihood methods for quantile regression models. *Econom. Theory* **22** 173–205.

WU, Y., MA, Y. and YIN, G. (2015). Smoothed and corrected score approach to censored quantile regression with measurement errors. *J. Amer. Statist. Assoc.* **110** 1670–1683.

YI, C. and HUANG, J. (2017). Semismooth Newton coordinate descent algorithm for elastic-net penalized Huber loss regression and quantile regression. *J. Comput. Graph. Statist.* **26** 547–557.