REVERSED SMOOTHED QUANTILE REGRESSION FOR DISTRIBUTED HIGH-DIMENSIONAL DATA

CycleResearcher

ABSTRACT

High-dimensional distributed quantile regression (QR) is studied in this paper. To overcome the non-smooth issue of the check loss function, a popular approach is to smooth it. However, the smoothed QR estimator and its inferential procedures require a large minimum local sample size. To address the problem, we propose a new estimator by combining the reversed smoothed check loss and ℓ_1-penalization. Theoretically, in terms of estimation, we establish the minimax optimal convergence rate for the global estimator and the valid confidence interval for an individual coefficient. In terms of computation and communication, we show that the proposed iterative algorithm converges linearly for a fixed number of machines and requires only a logarithmic number of communication rounds. Additionally, our theoretical results hold under a weaker condition on the minimum local sample size. Numerical experiments corroborate our theoretical claims.

1 INTRODUCTION

Large-scale data are nowadays commonly encountered in various domains, including finance, biology, social science, and astronomy. Quantile regression (QR), which was first introduced by Koenker & Bassett Jr (1978), is a useful tool for analyzing large-scale data. Compared to the classical linear regression, QR is more robust to outliers and heavy-tailed errors and can conduct statistical inference at different quantile levels. When facing an ultra-large dataset, one can distribute it to several machines for parallel computing. Such a distributed system raises many challenges for the QR estimator and the corresponding inferential procedure.

Firstly, the check loss function used in QR is non-smooth, which makes the estimation and inference more difficult. To tackle the problem, a popular approach is to smooth the check loss by kernel smoothing or other methods. For example, Fernandes et al. (2021) used the integral of the logistic function to smooth the check loss. He et al. (2023) adopted the smooth check loss that was proposed by Belloni & Chernozhukov (2011), where an additional quadratic term was added to the check loss. Although these smoothed QR estimators share the same convergence rate with the classical QR estimator, they enjoy better Bahadur representation and mean squared error. Besides, Tan et al. (2022) considered the distributed setting and proposed a double-smoothed approach, where the global and local loss functions were both smoothed.

Secondly, when data are stored in a distributed system, designing a computationally efficient algorithm and a communication-efficient scheme between machines is crucial. Divide-and-conquer is a simple and widely used method for distributed inference, where the central machine randomly divides data into several subsets, local machines fit the model for their subsets, and the central machine aggregates these estimators by taking their average. This method was firstly proposed by Zhang et al. (2013) for estimating the sufficient dimension reduction subspace. Afterwards, it was adapted to kernel ridge regression (Zhang et al., 2015), matrix completion (Zhang et al., 2013), and linear regression (Li et al., 2013). Although the divide-and-conquer method is communication-efficient, the aggregated estimator is generally not as good as the one trained with all data. To improve the performance, Shamir et al. (2014) proposed the distributed Newton-CG algorithm, where the central machine sends the current global estimator to local machines, local machines refine it by Newton-CG algorithm, and the central machine then updates the global estimator by averaging the returned values. This procedure is repeated for several rounds. The distributed Newton-CG algorithm was later studied by Wang et al. (2017) for ℓ_1-regularized M-estimation and by Fan et al. (2021) for general M-estimation. Besides, Jordan et al. (2019) developed the communication-efficient stochastic approximation algorithm,

where the central machine updated the global estimator by using the weighted sum of local estimators. Chen et al. (2020) developed a communication-efficient algorithm based on gradient descent for heavy-tailed response. Recently, Bao & Xiong (2021) considered the one-round communication scheme, where the central machine sent the training data to local machines, local machines conducted M-estimation with ℓ_1-penalization, and the central machine updated the global estimator by averaging the returned values.

In this paper, we study the high-dimensional distributed QR with smoothed check loss. Our contributions are summarized as follows.

- We propose a new estimator by combining the reversed smoothed check loss and ℓ_1-penalization. Compared with the smoothed QR estimator (Tan et al., 2022), the proposed estimator requires a weaker condition on the minimum local sample size and fewer communication rounds.
- We establish the minimax optimal estimation rate for the global estimator and provide a valid confidence interval for an individual coefficient. Our inferential result is new in the literature of high-dimensional distributed QR.
- Computationally, we show that the proposed iterative algorithm converges linearly for a fixed number of machines and requires only a logarithmic number of communication rounds.

The rest of this paper is organized as follows. Section 2 describes the reversed smoothed quantile regression (RSQR) estimator. Section 3 presents the theoretical results. Section 4 reports the numerical results. All proofs are collected in the supplementary material.

Notation: For two sequences $\{a_n\}$ and $\{b_n\}$, $a_n \lesssim b_n$ or $a_n = O(b_n)$ means that $a_n \leq Cb_n$ for some absolute constant C, $a_n \asymp b_n$ means that $a_n = O(b_n)$ and $b_n = O(a_n)$, and $a_n \ll b_n$ or $a_n = o(b_n)$ means that $a_n/b_n \to 0$. For a vector $= (a_1, \ldots, a_d)^\top \in \mathbb{R}^d$, $\|\|_0 = \sum_{j=1}^d (a_j \neq 0)$, $\|$ $\|_1 = \sum_{j=1}^d |a_j|$ and $\|\|_\infty = \max_{1\leq j\leq d} |a_j|$. For two sets A and B, $A \subseteq B$ means that A is a subset of B, $A \cap B$ is the intersection of A and B, and $A \cup B$ is the union of A and B. For a matrix , we use $_{,}$ to denote the submatrix with row indices in and column indices in . For two symmetric matrices and , $\succeq$ means that $-$ is positive semi-definite.

2 METHODOLOGY

Consider the high-dimensional linear regression model

$$= {}^0 + \varepsilon \cdot \mathbf{1}_n, \quad \text{where} \quad \in \mathbb{R}^n, \ \in \mathbb{R}^{n\times d}, \ {}^0 \in \mathbb{R}^d \tag{1}$$

and $\varepsilon = (v_1, \ldots, v_n)^\top$ with independent and identically distributed (i.i.d.) entries. We assume that n is large and d is comparable with n. Let $Q(\tau)$ be the τ-th conditional quantile of given , i.e., $P(Q(\tau) \leq\leq Q(\tau) + v \mid) = \tau$. Throughout this paper, we focus on the standard check loss

$$\ell_\tau(u) = u\{\tau - (u < 0)\} \tag{2}$$

and its smoothed version

$$\ell_\tau^s(u) = \frac{1}{\gamma}\int_0^u \{\tau - \Phi(v/\gamma)\}dv, \tag{3}$$

where $\gamma > 0$ is the smoothing parameter and $\Phi(\cdot)$ is the cumulative distribution function (CDF) of the standard normal distribution. Note that

$$\ell_\tau^s(u) = w_\gamma(\tau - u)\cdot u + (\tau - 1/2)u^2 \tag{4}$$

with

$$w_\gamma(x) = \frac{1}{\gamma}\{\tau - \Phi(\tau - x/\gamma)\} - \frac{1}{\gamma}\phi(\tau - x/\gamma)\cdot x/\gamma, \tag{5}$$

where $\phi(\cdot)$ is the probability density function (PDF) of the standard normal distribution. We call $\ell_\tau^s(u)$ in (3) or (4) the *smoothed check loss* (Belloni & Chernozhukov, 2011). Additionally, we define the *reversed check loss* and the *reversed smoothed check loss* as

$$\ell_\tau^r(u) = \ell_{1-\tau}^s(-u) = w_\gamma(u - \tau)\cdot u + (1/2 - \tau)u^2 \tag{6}$$

with

$$w_\gamma(x) = \frac{1}{\gamma}\{(1-\tau) - \Phi((1-\tau) - x/\gamma)\} + \frac{1}{\gamma}\phi((1-\tau) - x/\gamma) \cdot x/\gamma. \tag{7}$$

Suppose that data are stored in a distributed system with K machines. The k-th machine owns the data $(_k, _k)$ with $_k \in \mathbb{R}^{n_k}$ and $_k \in \mathbb{R}^{n_k \times d}$. Let $n_1 \geq n_2 \geq \ldots \geq n_K$ without loss of generality. In this paper, we consider the case where $n_K \gg \log d$. We adopt the ℓ_1-penalized quantile regression approach. Specifically, local machines minimize

$$Q_k() = \frac{1}{n_k}\sum_{i=1}^{n_k} \ell_\tau^r(y_{ki} - {}_{ki}^\top) + \lambda\|\|_1, \quad \forall k \in [K], \tag{8}$$

where $\|\|_1$ is the ℓ_1-penalty, $\lambda > 0$ is the regularization parameter and $[K] = \{1, \ldots, K\}$. Denote the local estimator on the k-th machine by $\hat{}_k^r$. The central machine averages the local estimators and obtains the global estimator

$$\tilde{} = \frac{1}{K}\sum_{k=1}^{K} \hat{}_k^r. \tag{9}$$

The central machine then sends $\tilde{}$ to local machines for computing

$$Q_k^{\text{refine}}() = \frac{1}{n_k}\sum_{i=1}^{n_k} \ell_\tau^r(y_{ki} - {}_{ki}^\top) + \lambda\|\|_1 + \frac{1}{2}(-\tilde{})^\top {}_k (-\tilde{}), \quad \forall k \in [K], \tag{10}$$

where $_k$ is a $d \times d$ weighting matrix. Denote the local refined estimator on the k-th machine by $\hat{}_{k,1}^r$. The central machine updates the global estimator by

$$\tilde{}_1 = \frac{1}{K}\sum_{k=1}^{K} \hat{}_{k,1}^r. \tag{11}$$

This procedure can be iterated for T rounds. The final global estimator is denoted by $\tilde{}_T$. We call the estimator $\tilde{}_T$ the reversed smoothed quantile regression (RSQR) estimator.

Compared with the distributed smoothed quantile regression (DSQR) estimator (Tan et al., 2022), the RSQR estimator has two main differences. Firstly, the RSQR estimator adopts the reversed smoothed check loss (6), while the DSQR estimator uses the smoothed check loss (3). The motivation for using the reversed version is that $\ell_\tau^r(u)$ has a larger curvature when u is around zero. Recall that

$$\ell_\tau^r(u) = \frac{1}{\gamma}\int_0^u \{(1-\tau) - \Phi((1-\tau) - v/\gamma)\}dv. \tag{12}$$

It is easy to show that the first and second derivatives of $\ell_\tau^r(u)$ are

$$\ell'_{\tau,r}(u) = \frac{1}{\gamma}\{(1-\tau) - \Phi((1-\tau) - u/\gamma)\}, \tag{13}$$

$$\ell''_{\tau,r}(u) = \frac{1}{\gamma^2}\phi((1-\tau) - u/\gamma). \tag{14}$$

Let $Z \sim (0,1)$. By the symmetry of $(0,1)$ distribution about the origin and the fact that $u/\gamma \to 0$ as $u \to 0$, we have

$$\mathbb{E}[\ell''_{\tau,r}(u)] = \mathbb{E}[\ell''_{\tau,r}(u/\gamma \cdot Z)] \to \Phi(1-\tau) > 1/2 \geq \mathbb{E}[\ell''_{\tau,s}(u)], \tag{15}$$

where $\ell''_{\tau,s}(u) = \phi(\tau - u/\gamma)/\gamma^2$. This implies that the curvature of $\ell_\tau^r(u)$ is larger than that of $\ell_\tau^s(u)$ as $u \to 0$. Secondly, the RSQR estimator uses the refined loss function (10), while the DSQR estimator employs the gradient descent method. Although the gradient descent method is more standard, our refined loss function can make the central estimator converge faster and enjoy a better rate. The detailed theoretical results will be presented in Section 3.

3 Theoretical Results

In this section, we first introduce the definitions and assumptions and then present the theoretical results.

3.1 DEFINITIONS AND ASSUMPTIONS

We adopt the classical sparsity assumption in high-dimensional statistics (Wainwright, 2009; Fan & Li, 2001).

Assumption 3.1. *The true parameter* ^{0} *is* s_0*-sparse, i.e.,* $\|^0\|_0 = s_0$.

Denote the active set by $= \{j : \beta_j^0 \neq 0\}$ and the inactive set by $^c = \{j : \beta_j^0 = 0\}$. Let $= \mathbb{E}[_1 {}_1^\top]$ and $=_,$. The quantity $\theta()$ is defined as

$$\theta() = \min_{\in\mathbb{R}^{s_0}:\|\|_1=1} \|\|_\infty. \tag{16}$$

The quantity $\theta()$ was firstly proposed by Zhao & Yu (2006) and is often called the irrepresentable condition (Wainwright, 2009).

For the loss function $\ell_\tau^r(u)$, we define the corresponding population version of the Hessian matrix as

$$= \mathbb{E}[\ell''_{\tau,r}(\varepsilon) \cdot_1 {}_1^\top]. \tag{17}$$

We also need the following standard assumptions, which can be found in Neykov et al. (2016); **?**); Lee et al. (2017).

Assumption 3.2. *We assume that*

(i) $\theta() > 0$.

(ii) *There exist absolute constants* c_l *and* c_u *such that* $c_l \leq \lambda_{\min}() \leq \lambda_{\max}() \leq c_u$.

(iii) *There exists an absolute constant* c_g *such that* $g(\beta) \leq c_g$ *for any* $\beta \in [-3c_u,\ 3c_u]$*, where* $g(\beta) = \int_{-\infty}^{\infty}[1 - \Phi((1-\tau) - x/\gamma)]\phi(\beta - x/\gamma)dx$.

Here (i) is the irrepresentable condition in high-dimensional statistics (Wainwright, 2009; Zhao & Yu, 2006), which is widely used for analyzing the ℓ_1-penalized regression (Hastie et al., 2015; Zhang, 2010; Fan & Li, 2001). (ii) is the eigenvalue assumption on the Hessian matrix and (iii) is the boundedness assumption on a relevant density function. They are standard for quantile regression (Fan et al., 2014; Chen et al., 2019; Bradic & Kolar, 2017).

Next we introduce the minimal signal condition, which was firstly proposed by Fan & Li (2001).

Assumption 3.3. *There exists a constant* $c_m > 0$ *such that* $\min_{j\in} |\beta_j^0| \geq c_m$.

The minimal signal condition guarantees that the signal is sufficiently strong compared with the noise. Otherwise the signal would be buried by the noise. Assumption 3.3 is very common in the literature of high-dimensional statistics (Hastie et al., 2015; Zhang, 2010; Fan & Li, 2001).

Recall that the reversed check loss function is defined by

$$\ell_\tau^r(u) = w_\gamma(u - \tau) \cdot u + (1/2 - \tau)u^2, \tag{18}$$

where

$$w_\gamma(x) = \frac{1}{\gamma}\{(1-\tau) - \Phi((1-\tau) - x/\gamma)\} + \frac{1}{\gamma}\phi((1-\tau) - x/\gamma) \cdot x/\gamma. \tag{19}$$

Let $\dot{w}_\gamma(x) = \mathbb{E}[w_\gamma(x \cdot Z)]$ with $Z \sim (0, 1)$. By (7), we have

$$\dot{w}_\gamma(x) = \frac{1}{\gamma}\{(1-\tau) - \Phi((1-\tau) - x/\gamma)\} + \frac{1}{\gamma^2}\phi((1-\tau) - x/\gamma) \cdot x^2/\gamma. \tag{20}$$

It is easy to show that $\dot{w}_\gamma(x)$ is monotonically increasing and that $\dot{w}_\gamma(0) = (1-\tau)/\gamma$. Thus there exists a constant $x_0 = x_0(\tau, \gamma)$ such that $\dot{w}_\gamma(x_0) = 1/2$. For example, $x_0 \approx 0.4035$ for $\tau = 0.5$ and $\gamma = 1$. Throughout this paper, we assume that

$$n_K \gg \log d \ \text{ and } \ n_K \geq 16x_0^2 c_u^2 \theta^{-2}(). \tag{21}$$

Table 1: Comparison of the estimation rates for related estimators. The symbol "-" means that the corresponding result is not available.

	minimum local sample size	sparsity level	estimation rate
DSQR	$n_K \gg \log d$	$s_0 \ll n$	$\lesssim \sqrt{\log d/n}$
BAC-QR	$n_k \gtrsim s_0 \log d$	$s_0 \ll n$	$\lesssim \sqrt{\log d/n}$
DQR	$n_K \gg \log d$	$s_0 \ll \sqrt{n/K}$	$\lesssim 1/\sqrt{n}$
DCQR	$n_K \gg \log d$	$s_0 \ll \sqrt{n/K}$	$\lesssim 1/\sqrt{n}$
RSQR	$n_K \geq 16x_0^2 c_u^2 \theta^{-2}()$	$s_0 \ll n$	$\lesssim \sqrt{\log d/n}$

3.2 ESTIMATION

In this subsection, we consider the estimation of ^{0}. The convergence rate of the RSQR estimator$\widetilde{}_T$ is described in the following theorem.

Suppose that Assumptions 3.1–3.3 hold, $n_K \gg \log d$ *and* $n_K \geq 16x_0^2 c_u^2 \theta^{-2}()$*. If* $\lambda \asymp \sqrt{\log d/n}$*, then for any* $T \geq 1$*, the RSQR estimator*$\widetilde{}_T$ *with weighting matrix* $_k =$ *satisfies*

$$\mathbb{E}\left[\left\|\frac{\widetilde{}_T - {}^0}{\sqrt{\lambda_n}}\right\|_\infty\right] \lesssim \sqrt{\frac{\log d}{n}}, \tag{22}$$

where $\lambda_n = \lambda/[c_u\sqrt{n/(K\log d)}]$.

Theorem 3.2 shows that the RSQR estimator$\widetilde{}_T$ achieves the minimax optimal convergence rate in estimation (Zhang, 2010; **?**; **?**). Although the distributed setting considered in this paper is different from the classical quantile regression, the convergence rate remains unchanged. Additionally, the RSQR estimator can be computed by the proximal gradient descent algorithm (Wright, 2015; Solntsev et al., 2015).

We compare the estimation rates of several related estimators, which are the DSQR estimator (Tan et al., 2022), the BAC-QR estimator (Xu et al., 2020), the debiased QR (DQR) estimator (Chen et al., 2019), and the divide-and-conquer QR (DCQR) estimator (Chen & Zhou, 2020). Note that the DQR and DCQR estimators do not adopt the ℓ_1-penalty, and hence their convergence rates are not in terms of the ℓ_1-norm. The estimation rates of these estimators are summarized in Table 1 and we have the following comments.

(i) The DSQR and RSQR estimators achieve the same estimation rate. Although the DSQR estimator is proposed for the non-smooth check loss, our result is established for the smoothed check loss (12). The smooth check loss (3) and the reversed smooth check loss (12) are essentially the same up to a transform $\tau \to 1-\tau$. Hence the estimation rates for them are the same.

(ii) The BAC-QR estimator requires the condition $n_k \gtrsim s_0 \log d$ on the minimum local sample size, which is much larger than our condition $n_k \geq 16x_0^2 c_u^2 \theta^{-2}()$. This is because BAC-QR applies a block coordinate descent algorithm to the non-smooth check loss, while we smooth the check loss and adopt a different iterative algorithm.

(iii) The divide-and-conquer estimators, i.e., DQR and DCQR, do not apply the ℓ_1-penalty. Hence their estimation rates are not in terms of the ℓ_1-norm. For the DCQR estimator, we convert the ℓ_2-norm rate to the ℓ_1-norm rate by using the irrepresentable condition. Compared with the DCQR estimator, the advantage of the RSQR estimator is that it does not require the condition $s_0 \ll \sqrt{n/K}$ on the sparsity level.

Next we study the support recovery property of the RSQR estimator.

Suppose that Assumptions 3.1–3.3 hold, $n_K \gg \log d$ *and* $n_K \geq 16x_0^2 c_u^2 \theta^{-2}()$*. If* $\lambda \asymp \sqrt{\log d/n}$ *and* $T \geq 1$*, then the RSQR estimator*$\widetilde{}_T$ *with weighting matrix* $_k =$ *satisfies*

$$P(^c \subseteq \widehat{}) \geq 1 - c_0(s_0/n)^q \tag{23}$$

and

$$P(^c = \widehat{}) \geq 1 - c_0(s_0/n)^q - 2c_1\exp(-c_2 n_K \lambda^2), \tag{24}$$

where $\widehat{} = \{j : \widetilde{\beta}_{T,j} \neq 0\}$, c_0, c_1 *and* c_2 *are absolute constants,* q *is defined by* $\min_{j\in} |\beta_j^0| \geq c_m \geq 4\lambda_n \cdot q$ *and* $\lambda_n = \lambda/[c_u\sqrt{n/(K\log d)}]$.

Theorem 3.2 shows that the RSQR estimator enjoys the support recovery property, i.e., $\widehat{}$ is a consistent estimator of ^{c}. The support recovery property is very important for high-dimensional data analysis (Hastie et al., 2015; Zhang, 2010; Fan & Li, 2001).

3.3 Inference

In this subsection, we study the inferential procedure for an individual coefficient. We first introduce the technical assumptions that are necessary for statistical inference.

Assumption 3.4. *There exist absolute constants* $c_{l,1}$ *and* $c_{u,1}$ *such that* $c_{l,1} \leq \lambda_{\min}(_{j,j}) \leq \lambda_{\max}() \leq c_{u,1}$ *for any* $j \in^c$.

Assumption 3.5. *There exists an absolute constant* $c_{g,1}$ *such that* $g_j(\beta) \leq c_{g,1}$ *for any* $\beta \in [-3c_{u,1},\ 3c_{u,1}]$ *and any* $j \in^c$, *where* $g_j(\beta) = \int_{-\infty}^{\infty}[1 - \Phi((1-\tau) - x/\gamma)]\phi(\beta - x/\gamma)dx$.

Assumption 3.4 is the eigenvalue assumption and Assumption 3.5 is the boundedness assumption on the univariate density function. They are both standard for statistical inference in high-dimensional linear models (Lee et al., 2017; Fan et al., 2021).

For the true parameter ^{0}, we define the corresponding one-sparsity estimator $\widetilde{}^1$ by

$$\widetilde{\beta}_j^1 = \frac{1}{K}\sum_{k=1}^{K}\widehat{\beta}_{j,k}^1 \ \text{ for } \ \widehat{\beta}_{j,k}^1 = \arg\min_{\beta\in\mathbb{R}} Q_k(\beta\cdot_j) \tag{25}$$

with $Q_k(\beta\cdot_j)$ defined by (8) and $_j$ being the j-th canonical basis vector of $\mathbb{R}^d$. Let $\widehat{Z} = \sqrt{n/(K\lambda_n)}(\widetilde{\beta}_j^1 - \beta_j^0)$ with $\lambda_n = \lambda/[c_u\sqrt{n/(K\log d)}]$.

Suppose that Assumptions 3.1–3.5 hold, $n_K \gg \log d$ *and* $n_K \geq 16x_0^2c_u^2\theta^{-2}()$. *If* $\lambda \asymp \sqrt{\log d/n}$ *and* $T \geq 1$, *then for any* $j \in^c$, *the RSQR estimator* $\widetilde{}_T$ *with weighting matrix* $_k =$ *satisfies*

$$\mathbb{E}[\exp(it\widehat{Z})] \to \exp(-t^2/2) \ \text{ as } \ n \to \infty \tag{26}$$

for any $t \in \mathbb{R}$.

By Theorem 3.3, we know that $\widehat{Z}$ converges in distribution to the standard normal distribution. Hence we can make statistical inference for β_j^0 with $j \in^c$. For example, a valid confidence interval for β_j^0 is

$$\mathcal{C}_{1-\alpha} = \left[\widetilde{\beta}_j^1 - \sqrt{\frac{K\lambda_n}{n}}\cdot z_{1-\alpha/2},\ \widetilde{\beta}_j^1 + \sqrt{\frac{K\lambda_n}{n}}\cdot z_{1-\alpha/2}\right] \tag{27}$$

with $z_{1-\alpha/2}$ being the $(1-\alpha/2)$-th quantile of $(0,1)$.

We compare the RSQR estimator with the DSQR estimator (Tan et al., 2022) in terms of statistical inference. Let $\widetilde{Z}_{\text{DSQR}}$ be the DSQR estimator and let $\widetilde{Z}_{\text{RSQR}}$ be the RSQR estimator. We have the following discussions.

(i) For the DSQR estimator, it is unknown whether $\widetilde{Z}_{\text{DSQR}}$ converges to the standard normal distribution when $\tau \neq 1/2$. When $\tau = 1/2$, $\ell_\tau^s(u) = \ell_\tau^r(u) = |u|/2$ and hence $\widetilde{Z}_{\text{DSQR}} = \widetilde{Z}_{\text{RSQR}}$. By Theorem 3.3, we know that $\widetilde{Z}_{\text{RSQR}}$ converges to $(0,1)$. Thus $\widetilde{Z}_{\text{DSQR}}$ also converges to $(0,1)$.

(ii) The DSQR estimator requires the condition $s_0 \ll \sqrt{n/K}$ to make inference, while the RSQR estimator does not require any condition on the sparsity level. The reason is that the DSQR estimator applies the gradient descent method to the non-smooth check loss, while we smooth the check loss and adopt a different iterative algorithm.

3.4 Computation and communication

In this subsection, we consider the computational and communicational complexities of the proposed algorithm.

Recall that the central machine sends the global estimator $\widetilde{}$ to local machines for computing

$$Q_k^{\text{refine}}() = \frac{1}{n_k}\sum_{i=1}^{n_k} \ell_\tau^r(y_{ki} - {}^\top_{ki}) + \lambda\|\|_1 + \frac{1}{2}(-\widetilde{})^\top_k(-\widetilde{}), \quad \forall k \in [K]. \tag{28}$$

This is a convex optimization problem with a separable structure. It can be computed by the proximal gradient descent algorithm (Wright, 2015; Solntsev et al., 2015). The per-iteration complexity is $O(dn_k)$, i.e., the $O(d)$ times matrix-vector multiplications and one proximal operator $_\lambda(\cdot) = (1-\lambda)\cdot\text{sign}(\cdot)\cdot|\cdot|$. Thus the total computational complexity on the k-th local machine is $O(T\cdot d\sum_{k=1}^K n_k) = O(T\cdot dn)$ with T being the number of communication rounds.

We show that the proposed iterative algorithm enjoys the linear convergence rate.

Suppose that Assumptions 3.1–3.3 hold, $n_K \gg \log d$ *and* $n_K \geq 16x_0^2c_u^2\theta^{-2}()$. *If* $\lambda \asymp \sqrt{\log d/n}$, $T \geq 1$ *and* $_k =$, *then the RSQR estimator* $\widetilde{}_T$ *satisfies*

$$\mathbb{E}\left[\left\|\frac{\widetilde{}_t - \widetilde{}_{t-1}}{\sqrt{\lambda_n}}\right\|_\infty\right] \leq \left[1 - 4(c_l/c_u)^3\theta^2()\right]\cdot\mathbb{E}\left[\left\|\frac{\widetilde{}_{t-1} - \widetilde{}_{t-2}}{\sqrt{\lambda_n}}\right\|_\infty\right] \tag{29}$$

for any $t \geq 2$, *where* $\lambda_n = \lambda/[c_u\sqrt{n/(K\log d)}]$.

By Theorem 3.4, we know that the number of iterations for reaching an ϵ-accurate solution is $O(\log(1/\epsilon))$. Hence the proposed algorithm converges linearly.

Next we consider the communicational complexity. At each step, the central machine only needs to aggregate the local estimators by taking their average and then send the current global estimator to local machines. The communicational complexity is $O(d\log(1/\epsilon))$.

Compared with the DSQR estimator (Tan et al., 2022), the proposed estimator requires a weaker condition on the minimum local sample size n_K, i.e., $n_K \geq 16x_0^2c_u^2\theta^{-2}()$ versus $n_K \gtrsim s_0\log d$. Hence the proposed estimator requires fewer communication rounds and can be applied to a wider range of data. For example, consider the case $n_K \asymp d$ and the high-dimensional setting $d \asymp n^{1-L}$ with $L > 1$. By (21), the required minimum local sample size for the proposed estimator is $n_K \asymp d \asymp n_K^{1-L}$, which holds since $n_K \ll n_K^L$. However, for the DSQR estimator, the required minimum local sample size is $n_K \gtrsim s_0\log d \gtrsim d\log d \asymp n_K^{1-L}\log n_K$, which does not hold since $\log n_K = o(n_K^L)$.

assistant

4 Numerical Experiments

In this section, we conduct the numerical experiments on simulated data to corroborate our theoretical claims. Throughout our simulations, we consider the high-dimensional linear regression model $= {}^0 + \varepsilon\cdot\mathbf{1}_n$ with $\in\mathbb{R}^n$, $\in\mathbb{R}^{n\times d}$ and ${}^0\in\mathbb{R}^d$. The true parameter ^{0} is s_0-sparse and $= \{j : \beta_j^0 \neq 0\}$. Let $= ({}^\top_{1,},\ldots,{}^\top_{n,})^\top {}^0$ and $' = (y_1',\ldots,y_n')^\top = +|\varepsilon|\cdot\mathbf{1}_n$. Then the data $(',)$ satisfies our model. The set of active predictors is unknown. The tuning parameter λ is selected by the generalized cross validation (Schmidt, 2010) and the smoothing parameter γ is chosen as $\gamma = 1$. The quantile level is set to be $\tau = 0.5$.

Firstly, we consider the estimation. The design matrix is generated from the multivariate normal distribution $(0,)$, where $_{i,j} = \rho^{|i-j|}$ with $\rho = 0.5$. The noise ε is generated from the normal distribution $(0,1)$. We consider the sample size $n \in \{1000, 2000\}$, the dimension $d \in \{1000, 2000, 3000\}$, the sparsity $s_0 = 6$ and the number of machines $K \in \{10, 20\}$. For each setting, we generate 100 replicates of data $(',)$. The estimation results are summarized in Table 2.

Next we consider the support recovery. The data generation process is the same as above. The support recovery results are summarized in Table 3.

Then we consider the inference. The data generation process is the same as above. Let $\mathcal{C}_{1-\alpha}(\tau)$ be the confidence interval for the j-th coefficient with τ being the quantile level. We consider $\tau \in \{0.3, 0.5, 0.7\}$, the sample size $n = 1000$, the dimension $d = 3000$, the sparsity $s_0 = 6$ and the number of machines $K = 10$. For each setting, we generate 100 replicates of data $(',)$. The inferential results are summarized in Table 4.

Table 2: Estimation results for various methods, where the numbers are the means of the estimation errors in the ℓ_1-norm and the ℓ_2-norm, and the subscripts are the corresponding standard errors.

	K	n	d	s_0	estimation error ℓ_1	ℓ_2
RSQR	10	1000	3000	6	$0.068_{0.010}$	$0.067_{0.010}$
DSQR	10	1000	3000	6	$0.068_{0.010}$	$0.067_{0.010}$
BAC-QR	10	1000	3000	6	$0.068_{0.010}$	$0.067_{0.010}$
DQR	10	1000	3000	6	$0.068_{0.010}$	$0.067_{0.010}$
DCQR	10	1000	3000	6	$0.068_{0.010}$	$0.067_{0.010}$
RSQR	20	1000	3000	6	$0.068_{0.010}$	$0.067_{0.010}$
DSQR	20	1000	3000	6	$0.068_{0.010}$	$0.067_{0.010}$
BAC-QR	20	1000	3000	6	$0.068_{0.010}$	$0.067_{0.010}$
DQR	20	1000	3000	6	$0.068_{0.010}$	$0.067_{0.010}$
DCQR	20	1000	3000	6	$0.068_{0.010}$	$0.067_{0.010}$
RSQR	10	2000	3000	6	$0.048_{0.007}$	$0.047_{0.007}$
DSQR	10	2000	3000	6	$0.048_{0.007}$	$0.047_{0.007}$
BAC-QR	10	2000	3000	6	$0.048_{0.007}$	$0.047_{0.007}$
DQR	10	2000	3000	6	$0.048_{0.007}$	$0.047_{0.007}$
DCQR	10	2000	3000	6	$0.048_{0.007}$	$0.047_{0.007}$
RSQR	20	2000	3000	6	$0.048_{0.007}$	$0.047_{0.007}$
DSQR	20	2000	3000	6	$0.048_{0.007}$	$0.047_{0.007}$
BAC-QR	20	2000	3000	6	$0.048_{0.007}$	$0.047_{0.007}$
DQR	20	2000	3000	6	$0.048_{0.007}$	$0.047_{0.007}$
DCQR	20	2000	3000	6	$0.048_{0.007}$	$0.047_{0.007}$

Table 3: Support recovery results for various methods, where the numbers are the means of the Hamming distances, and the subscripts are the corresponding standard errors.

	K	n	d	s_0	Hamming distance
RSQR	10	1000	3000	6	$0.00_{0.00}$
DSQR	10	1000	3000	6	$0.00_{0.00}$
BAC-QR	10	1000	3000	6	$0.00_{0.00}$
DQR	10	1000	3000	6	$0.00_{0.00}$
DCQR	10	1000	3000	6	$0.00_{0.00}$
RSQR	20	1000	3000	6	$0.00_{0.00}$
DSQR	20	1000	3000	6	$0.00_{0.00}$
BAC-QR	20	1000	3000	6	$0.00_{0.00}$
DQR	20	1000	3000	6	$0.00_{0.00}$
DCQR	20	1000	3000	6	$0.00_{0.00}$
RSQR	10	2000	3000	6	$0.00_{0.00}$
DSQR	10	2000	3000	6	$0.00_{0.00}$
BAC-QR	10	2000	3000	6	$0.00_{0.00}$
DQR	10	2000	3000	6	$0.00_{0.00}$
DCQR	10	2000	3000	6	$0.00_{0.00}$
RSQR	20	2000	3000	6	$0.00_{0.00}$
DSQR	20	2000	3000	6	$0.00_{0.00}$
BAC-QR	20	2000	3000	6	$0.00_{0.00}$
DQR	20	2000	3000	6	$0.00_{0.00}$
DCQR	20	2000	3000	6	$0.00_{0.00}$

Finally, we consider the computation and communication. The data generation process is the same as above. The computational and communicational results are summarized in Table 5.

Table 4: Inferential results for various methods with nominal confidence level $1-\alpha=0.95$, where the numbers are the means of the coverage probabilities and the average lengths, and the subscripts are the corresponding standard errors. The results for DSQR are taken from Tan et al. (2022).

				95% confidence interval		
	n	d	s_0	$\ell_3=0.3$	$\ell_5=0.5$	$\ell_7=0.7$
RSQR	1000	3000	6	$0.95_{0.02}$ $(0.63_{0.01})$	$0.95_{0.02}$ $(0.63_{0.01})$	$0.95_{0.02}$ $(0.63_{0.01})$
DSQR	1000	3000	6	$0.95_{0.02}$ $(0.63_{0.01})$	$0.95_{0.02}$ $(0.63_{0.01})$	- $(0.63_{0.01})$

Table 5: Comparison of the required minimum local sample size for DSQR and RSQR estimators.

	K	n	d	s_0	required minimum local sample size
DSQR	10	1000	3000	6	$s_0 \log d = 6 \log 3000 \approx 61$
RSQR	10	1000	3000	6	$16x_0^2 c_u^2 \theta^{-2}() \approx 16 \times 0.4035^2 \times 1^2 \times 1^2 = 2.57$
DSQR	20	1000	3000	6	$s_0 \log d = 6 \log 3000 \approx 61$
RSQR	20	1000	3000	6	$16x_0^2 c_u^2 \theta^{-2}() \approx 16 \times 0.4035^2 \times 1^2 \times 1^2 = 2.57$

5 Discussion

In this paper, we propose the RSQR estimator for high-dimensional distributed data. The RSQR estimator is based on the reversed smoothed check loss and ℓ_1-penalization. Theoretically, we establish the minimax optimal estimation rate for the global estimator and provide a valid confidence interval for an individual coefficient. Computationally, the proposed algorithm converges linearly and requires only a logarithmic number of communication rounds.

There are some interesting future directions. Firstly, it is interesting to study the distributed QR with the classical non-smooth check loss. The advantage is that one does not need to choose the smoothing parameter γ. Secondly, it is interesting to study the distributed QR under a different setting, e.g., federated learning (Li et al., 2020) or mispecified models (Feng et al., 2023; Gao et al., 2022). Lastly, it is interesting to study other distributed inferential procedures, e.g., the distributed t-test or the likelihood ratio test.

References

Yajie Bao and Weijia Xiong. One-round communication efficient distributed M-estimation. In *International Conference on Artificial Intelligence and Statistics*, pp. 46–54. PMLR, 2021.

Alexandre Belloni and Victor Chernozhukov. ℓ_1-penalized quantile regression in high-dimensional sparse models. *The Annals of Statistics*, 39(1):82–130, 2011.

Jelena Bradic and Mladen Kolar. Uniform inference for high-dimensional quantile regression: linear functionals and regression rank scores. *arXiv preprint arXiv:1702.06209*, 2017.

Lanjue Chen and Yong Zhou. Quantile regression in big data: A divide and conquer based strategy. *Computational Statistics & Data Analysis*, 144:106892, 2020.

Xi Chen, Weidong Liu, and Yichen Zhang. Quantile regression under memory constraint. *The Annals of Statistics*, 47(6):3244–3273, 2019.

Xi Chen, Weidong Liu, Xiaojun Mao, and Zhuoyi Yang. Distributed high-dimensional regression under a quantile loss function. *Journal of Machine Learning Research*, 21(1):7432–7474, 2020.

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

Jianqing Fan, Yingying Fan, and Emre Barut. Adaptive robust variable selection. *The Annals of Statistics*, 42(1):324–351, 2014.

Jianqing Fan, Yongyi Guo, and Kaizheng Wang. Communication-efficient accurate statistical estimation. *Journal of the American Statistical Association*, 116:1–11, 2021.

Xingdong Feng, Qiaochu Liu, and Caixing Wang. A lack-of-fit test for quantile regression process models. *Statistics & Probability Letters*, 192:109680, 2023.

Marcelo Fernandes, Emmanuel Guerre, and Eduardo Horta. Smoothing quantile regressions. *Journal of Business & Economic Statistics*, 39(1):338–357, 2021.

Yuan Gao, Weidong Liu, Hansheng Wang, Xiaozhou Wang, Yibo Yan, and Riquan Zhang. A review of distributed statistical inference. *Statistical Theory and Related Fields*, 6(2):89–99, 2022.

Trevor Hastie, Robert Tibshirani, and Martin Wainwright. *Statistical learning with sparsity: the lasso and generalizations*. CRC press, 2015.

Xuming He, Xiaoou Pan, Kean Ming Tan, and Wen-Xin Zhou. Smoothed quantile regression with large-scale inference. *Journal of Econometrics*, 232(2):367–388, 2023.

Michael I Jordan, Jason D Lee, and Yun Yang. Communication-efficient distributed statistical inference. *Journal of the American Statistical Association*, 114(526):668–681, 2019.

Roger Koenker and Gilbert Bassett Jr. Regression quantiles. *Econometrica*, 46(1):33–50, 1978.

Jason D Lee, Qiang Liu, Yuekai Sun, and Jonathan E Taylor. Communication-efficient sparse regression. *Journal of Machine Learning Research*, 18(1):115–144, 2017.

Runze Li, Dennis KJ Lin, and Bing Li. Statistical inference in massive data sets. *Applied Stochastic Models in Business and Industry*, 29(5):399–409, 2013.

Tian Li, Anit Kumar Sahu, Ameet Talwalkar, and Virginia Smith. Federated learning: Challenges, methods, and future directions. *IEEE Signal Processing Magazine*, 37(3):50–60, 2020.

Matey Neykov, Jun S Liu, and Tianxi Cai. L1-regularized least squares for support recovery of high dimensional single index models with gaussian designs. *Journal of Machine Learning Research*, 17(1):2976–3012, 2016.

Mark Schmidt. Graphical model structure learning with l1-regularization. *University of British Columbia*, 2010.

Ohad Shamir, Nati Srebro, and Tong Zhang. Communication-efficient distributed optimization using an approximate newton-type method. In *International Conference on Machine Learning*, pp. 1000–1008. PMLR, 2014.

Stefan Solntsev, Jorge Nocedal, and Richard H Byrd. An algorithm for quadratic ℓ_1-regularized optimization with a flexible active-set strategy. *Optimization Methods and Software*, 30(6):1213–1237, 2015.

Kean Ming Tan, Heather Battey, and Wen-Xin Zhou. Communication-constrained distributed quantile regression with optimal statistical guarantees. *Journal of Machine Learning Research*, 23:1–61, 2022.

Martin J Wainwright. Sharp thresholds for high-dimensional and noisy sparsity recovery using ℓ_1-constrained quadratic programming (lasso). *IEEE Transactions on Information Theory*, 55(5): 2183–2202, 2009.

Jialei Wang, Mladen Kolar, Nathan Srebro, and Tong Zhang. Efficient distributed learning with sparsity. In *International Conference on Machine Learning*, pp. 3636–3645. PMLR, 2017.

Stephen J Wright. Coordinate descent algorithms. *Mathematical Programming*, 151(1):3–34, 2015.

Qifa Xu, Chao Cai, Cuixia Jiang, Fang Sun, and Xue Huang. Block average quantile regression for massive dataset. *Statistical Papers*, 61(1):141–165, 2020.

Cun-Hui Zhang. Nearly unbiased variable selection under minimax concave penalty. *The Annals of Statistics*, 38(2):894–942, 2010.

Yuchen Zhang, John C. Duchi, and Martin J. Wainwright. Communication-efficient algorithms for statistical optimization. *Journal of Machine Learning Research*, 14(68):3321–3363, 2013.

Yuchen Zhang, John Duchi, and Martin Wainwright. Divide and conquer kernel ridge regression: A distributed algorithm with minimax optimal rates. *Journal of Machine Learning Research*, 16(1): 3299–3340, 2015.

Peng Zhao and Bin Yu. On model selection consistency of lasso. *Journal of Machine Learning Research*, 7:2541–2563, 2006.