Journal of Machine Learning Research 21 (2020) 1-31 Submitted 4/20; Revised 10/20; Published 10/20

High-dimensional Quantile Tensor Regression

Wenqi Lu WENQILU4-C@MY.CITYU.EDU.HK
School of Management
Fudan University
Shanghai, China
and
Department of Mathematics
City University of Hong Kong
Hong Kong, China

Zhongyi Zhu ZHUZY@FUDAN.EDU.CN
School of Management
Fudan University
Shanghai, China

Heng Lian HENGLIAN@CITYU.EDU.HK
Department of Mathematics
City University of Hong Kong
Hong Kong, China

Editor: Animashree Anandkumar

Abstract

Quantile regression is an indispensable tool for statistical learning. Traditional quantile regression methods consider vector-valued covariates and estimate the corresponding coefficient vector. Many modern applications involve data with a tensor structure. In this paper, we propose a quantile regression model which takes tensors as covariates, and present an estimation approach based on Tucker decomposition. It effectively reduces the number of parameters, leading to efficient estimation and feasible computation. We also use a sparse Tucker decomposition, which is a popular approach in the literature, to further reduce the number of parameters when the dimension of the tensor is large. We propose an alternating update algorithm combined with alternating direction method of multipliers (ADMM). The asymptotic properties of the estimators are established under suitable conditions. The numerical performances are demonstrated via simulations and an application to a crowd density estimation problem.

Keywords: Multidimensional array, Quantile regression, Sparsity principle, Tensor regression, Tucker decomposition

1. Introduction

Quantile regression (Koenker and Bassett, 1978) provides a useful approach to analyse the heterogeneous impact of regressors on different parts of the conditional distribution of the response, exhibits robustness to outliers, comes with well-developed computational algorithms, and thus is widely used in applications (Koenker, 2005). There is a large literature on the computational aspects and the asymptotic theories of quantile regression in both low-dimensional and high-dimensional settings, see, for example, Koenker (2005); Belloni and Chernozhukov (2011); Yu et al. (2017); Yi and Huang (2017). Most of the existing works focused on scenarios in which the covariates and parameters of interest are vectors. However, in fields like image/video analysis or recommendation systems (Zhou et al., 2018; Rendle and Schmidt-Thieme, 2010; Liu et al., 2013), data often take the form of multidimensional arrays, also known as tensors. For example, in the real data application part, we take a $136 \times 186 \times 3$ tensor extracted from surveillance video image in crowd dataset PETS2009 (Ferryman and Shahrokni, 2009) as the covariate. Using it as the predictor in a regression, the number of parameters is $136 \times 186 \times 3 = 75888$. Moreover, vectorizing procedure, which destroys the spatial structure of an image, would result in a loss of information.

In statistical models involving tensors, different types of tensor decomposition techniques are almost always used to treat tensor variables with a reduced number of parameters (Kolda and Bader, 2009; Chi and Kolda, 2012; Liu et al., 2012; Anandkumar et al., 2014a,b; Sun et al., 2017). Tensor methods can also be applied to deep learning models, aiming to reduce the number of parameters by imposing low-dimensional structures on tensors. For example, by applying tensor decomposition to neural network layers, it can be used for network model compression (Novikov et al., 2015; Kolbeinsson et al., 2019). Castellana and Bacciu (2020) used tensor methods in constructing the aggregation functions in LSTM (long short-term memory). Su et al. (2020) proposed a convolutional tensor-train LSTM for spatio-temporal learning. In some other works, converting input vectors in traditional machine learning and statistical models to tensors leads to new tensor-based methods, such as tensor clustering (Sun and Li, 2019) and vector autoregressive time series (Wang et al., 2019).

Coming to the regression problems, there have been a sequence of developments concerning mean regression with tensor predictors, tensor responses and/or tensor parameters. Several papers have studied regression with tensor response, see for example Rabusseau and Kadri (2016); Li and Zhang (2017); Sun and Li (2017). For scalar on tensor regression, there are considerably more works, some dealing with high-dimensional cases, based on different decomposition methods. Guo et al. (2012) proposed tensor ridge regression and support tensor regression based on CANDECOMP/PARAFAC (CP) decomposition. Zhou et al. (2013) proposed an estimation procedure for general linear tensor regression model which also used CP decomposition and studied its asymptotic properties. Li et al. (2018) applied a more flexible decomposition, called Tucker decomposition (Kolda and Bader, 2009), to the same regression model as Zhou et al. (2013). Tucker decomposition decomposes the coefficient tensor into a core tensor multiplied by a factor matrix along each mode. It includes CP as a special case where the core tensor is diagonal and the ranks in different modes are equal. As pointed out in Li et al. (2018), it has advantages in accommodating tensors with skewed dimensions and allows explicit modelling of interactions compared to CP decomposition. Apart from CP and Tucker decomposition, Liu et al. (2020) applied Tensor-Train (TT) decomposition (Oseledets, 2011) to a tensor on tensor regression model. TT decomposition was chosen because it has advantages in high order tensors. Alternative to using tensor decompositions which typically results in non-convex optimization problems, Raskutti et al. (2019) considered convex regularization techniques to exploit low-rank and sparse properties in tensor regression. Tensor regression can also be incorporated as trainable layers in deep neural networks (Cao and Rabusseau, 2017; Kossaifi et al., 2020).

While all the literature mentioned above tackle mean regression, quantile regression with tensor covariates is rarely studied. Since quantile regression has advantages over mean regression when there are outliers or the distribution of response is skewed, and it can be used to build prediction intervals, many classical machine learning tools have been generalized to conditional quantiles, for example neural networks and random forests (Taylor, 2000; Meinshausen, 2006). Our goal is to fill the void in quantile tensor regression and provide estimation methods for this problem.

In this paper, we propose methodologies to estimate the tensor coefficient in quantile regression. We assume the regression coefficient tensor has a low-rank structure, adopting Tucker decomposition to reduce the dimensionality to a manageable level, resulting in a parsimonious model. Moreover, we develop an alternating update algorithm for the proposed estimator and establish its asymptotic normality under some conditions.

Furthermore, we introduce a sparse Tucker decomposition for the coefficient tensor. Sparsity assumption is commonly used in high-dimensional statistical problems. When the dimensions are large, it is reasonable to incorporate sparsity to further reduce the number of parameters. For instance, Raskutti et al. (2019) discussed several scenarios in which sparsity occurs at entry-wise, fiber-wise or slice-wise level of a tensor, and presented a general convex regularized optimization approach. Here we assume there exists a Tucker decomposition such that the factor matrices are sparse orthogonal matrices. We induce sparsity by penalizing the Kronecker product of factor matrices using ℓ_1 penalty. For the sparse tensor quantile regression, we use alternating update combined with ADMM algorithm to deal with the ℓ_1 penalty and orthogonality constraints, and derive an upper bound of statistical estimation error.

Although conceptually straightforward, we make some important contributions which are summarized below.

- We extend the use of Tucker decomposition to quantile regression, which is an important tool in statistical analysis as demonstrated in the large literature.

- We establish the rates of convergence of both non-sparse and sparse quantile tensor regression estimator. Development of theoretical results for quantile regression using tensor decomposition is challenging, especially in high dimensions. Our proof indeed shows that it contains a lot more technical details compared to, say, quantile linear regression using lasso, or various least squares regression models.

- Computationally, for the case with sparse penalty, we propose an ADMM based algorithm, which is not needed in Li et al. (2018). Compared with prior works which applied ADMM to tensor decomposition (Zhang et al., 2014; Xie et al., 2018; Huang et al., 2016; Smith et al., 2017) and tensor regression (Wang et al., 2019), our algorithm is more complex due to the simultaneous use of quantile loss, sparsity penalty, and the orthogonality constraint, which requires introducing more auxiliary variables.

The rest of the paper is organized as follows. In Section 2, we introduce the tensor quantile regression model based on Tucker decomposition, and present the estimation and implementation details for both nonsparse and sparse scenarios. Section 3 establishes the theoretical properties. In Section 4, we investigate the finite sample properties via simulation studies, and Section 5 presents an application of the proposed method in crowd density estimation problem.

2. Models and estimation

2.1 Preliminaries

Tensors, or multidimensional arrays, are the fundamental constructs in our study. We start with a brief introduction to tensors and their decomposition. We refer readers to Kolda and Bader (2009) for a more detailed review.

Throughout this paper, we denote tensors by boldface script capital letters such as $\boldsymbol{\mathcal{X}}, \boldsymbol{\mathcal{Y}}$, matrices by boldface capital letters $\boldsymbol{X}, \boldsymbol{Y}$, and vectors by small boldface letters $\boldsymbol{x}, \boldsymbol{y}$. For a vector $\boldsymbol{x}$, let $\|\boldsymbol{x}\|_1$, $\|\boldsymbol{x}\|$ and $\|\mathbf{x}\|_\infty$ denote its ℓ_1, ℓ_2 and ℓ_∞ norms. For a matrix $\boldsymbol{X}$, its Frobenius norm, spectral norm, vectorization and jth largest singular value are denoted by $\|\boldsymbol{X}\|_F$, $\|\mathbf{X}\|_{op}$, $\text{vec}(\boldsymbol{X})$ and $\sigma_j(\boldsymbol{X})$, respectively. The ℓ_1 norm of $\boldsymbol{X}$ is $\|\boldsymbol{X}\|_1 = \|\text{vec}(\boldsymbol{X})\|_1$. The Kronecker product of two matrices $\boldsymbol{X}$ and $\boldsymbol{Y}$ is denoted by $\boldsymbol{X} \otimes \boldsymbol{Y}$. For a tensor $\boldsymbol{\mathcal{X}} \in \mathbb{R}^{p_1 \times \cdots \times p_K}$, its Frobenius norm and inner product are $\|\boldsymbol{\mathcal{X}}\|_F = \sqrt{\sum_{i_1=1}^{p_1} \sum_{i_2=1}^{p_2} \cdots \sum_{i_K=1}^{p_K} x_{i_1 i_2 \ldots i_k}^2}$ and $\langle \boldsymbol{\mathcal{X}}, \boldsymbol{\mathcal{Y}} \rangle = \sum_{i_1=1}^{p_1} \sum_{i_2=1}^{p_2} \cdots \sum_{i_K=1}^{p_K} x_{i_1 i_2 \ldots i_k} y_{i_1 i_2 \ldots i_k}$, respectively.

The order of a tensor is the number of dimensions, a.k.a modes— a multidimensional array $\boldsymbol{\mathcal{A}} \in \mathbb{R}^{p_1 \times \ldots, p_K}$ is called a Kth-order tensor. The matricization of a tensor links the concepts and properties of matrices to tensors. Mode-k matricization of a tensor $\boldsymbol{\mathcal{A}} \in \mathbb{R}^{p_1 \times \cdots \times p_K}$, denoted by $\boldsymbol{\mathcal{A}}_{(k)}$, arranges the mode-$k$ fibers (all p_k-dimensional vectors obtained by fixing all indices of the tensor $\boldsymbol{\mathcal{A}}$ except for the k-th index) to be the columns of the resulting matrix. For example, the mode-1 matricization of $\boldsymbol{\mathcal{A}} \in \mathbb{R}^{p_1 \times p_2 \times p_3}$ is the $p_1 \times (p_2 p_3)$ matrix $\boldsymbol{\mathcal{A}}_{(1)}$ such that

$$\{\boldsymbol{\mathcal{A}}_{(1)}\}_{i,(j-1)p_3+k} = \boldsymbol{\mathcal{A}}_{ijk}, \quad \forall 1 \leq i \leq p_1, 1 \leq j \leq p_2, 1 \leq k \leq p_3.$$

Then, we can define the vectorization of $\boldsymbol{\mathcal{A}}$ by $\text{vec}(\boldsymbol{\mathcal{A}}) = \text{vec}(\boldsymbol{\mathcal{A}}_{(1)})$. Tensors can be multiplied by matrices. The mode-k multiplication of tensor $\boldsymbol{\mathcal{A}} \in \mathbb{R}^{p_1 \times p_2 \times \cdots \times p_K}$ with a matrix $\boldsymbol{U} \in \mathbb{R}^{r_k \times p_k}$ is defined as

$$\left(\boldsymbol{\mathcal{A}} \times_k \boldsymbol{U}\right)_{i_1 \cdots i_{k-1} j i_{k+1} \cdots i_K} = \sum_{i_k=1}^{p_k} \boldsymbol{\mathcal{A}}_{i_1 i_2 \cdots i_K} \boldsymbol{U}_{j i_k}.$$

The definition of rank for a tensor $\boldsymbol{\mathcal{A}} \in \mathbb{R}^{p_1 \times \ldots, p_K}$ is not universal and many definitions of rank have been proposed in the literature. In this paper, we consider the multilinear ranks $(r_1, r_2, \ldots, r_K)$, where r_k is the dimension of vector space spanned by the mode-k fibers. This rank is related to Tucker decomposition. For a given tensor $\boldsymbol{\mathcal{A}}$ of rank $(r_1, r_2, \ldots, r_K)$, there exists a Tucker decomposition

$$\boldsymbol{\mathcal{A}} = \boldsymbol{\mathcal{G}} \times_1 \boldsymbol{U}_1 \times_2 \boldsymbol{U}_2 \cdots \times_K \boldsymbol{U}_K,$$

where $\boldsymbol{U}_k \in \mathbb{R}^{p_k \times r_k}$, $k = 1, \ldots, K$ are factor matrices which can be assumed to be orthogonal if so desired, and $\boldsymbol{\mathcal{G}} \in \mathbb{R}^{r_1 \times \cdots \times r_K}$ is the core tensor, whose entries are related to the level of interactions between factors. A special Tucker decomposition called higher-order singular value decomposition (HOSVD, Lathauwer et al., 2000) is often adopted. The factor matrix

$\boldsymbol{U}_k$ of HOSVD can be computed by the leading r_k left singular vectors of $\boldsymbol{\mathcal{A}}_{(k)}$ for each $k = 1, \ldots, K$, and the core tensor $\boldsymbol{\mathcal{G}} = \boldsymbol{\mathcal{A}} \times_1 \boldsymbol{U}_1^T \times_2 \boldsymbol{U}_2^T \cdots \times_K \boldsymbol{U}_K^T$. In this way, matrices $\boldsymbol{U}_k,\ k = 1, \ldots, K$ are orthogonal, and $\boldsymbol{\mathcal{G}}$ is all orthogonal, meaning the rows of each $\boldsymbol{\mathcal{G}}_{(k)},\ k = 1, \ldots, K$ are mutually orthogonal.

2.2 Tucker tensor quantile regression

In this paper, we consider quantile regression models with a scalar response y, and tensor covariate $\boldsymbol{\mathcal{X}} \in \mathbb{R}^{p_1 \times \cdots \times p_K}$. The τth conditional quantile of response given covariate $\boldsymbol{\mathcal{X}}$ is defined as $Q_\tau(y|\boldsymbol{\mathcal{X}}) = \inf\left\{t; F_{y|\boldsymbol{\mathcal{X}}}(t) \geq \tau\right\}$, where $F_{y|\boldsymbol{\mathcal{X}}}(t)$ is the condition distribution function of y given $\boldsymbol{\mathcal{X}}$. We consider a linear quantile regression model

$$Q_\tau(y|\boldsymbol{\mathcal{X}}) = \mu + \langle \boldsymbol{\mathcal{A}}, \boldsymbol{\mathcal{X}} \rangle,$$

where $\boldsymbol{\mathcal{A}}$ is a coefficient tensor of the same size as $\boldsymbol{\mathcal{X}}$ which captures the effect of each element in $\boldsymbol{\mathcal{X}}$. Since the intercept μ can be more easily dealt with both computationally and theoretically, in the following we will suppress the intercept μ for convenience of notation and explain briefly the minor modifications required in the algorithm and proofs at several places below. In this model, the number of parameters is $\prod_{k=1}^K p_k$, which is often large compared to the sample size. In order to reduce the number of parameters by exploiting the tensor structure, we impose a low-rank assumption on $\boldsymbol{\mathcal{A}}$. As mentioned before, if $\boldsymbol{\mathcal{A}}$ has multilinear rank $(r_1, r_2, \ldots, r_K)$, then there exists a Tucker decomposition

$$\boldsymbol{\mathcal{A}} = \boldsymbol{\mathcal{G}} \times_1 \boldsymbol{U}_1 \times_2 \boldsymbol{U}_2 \cdots \times_K \boldsymbol{U}_K,$$

where $\boldsymbol{\mathcal{G}} \in \mathbb{R}^{r_1 \times \cdots \times r_K}$ is the core tensor, and $\boldsymbol{U}_k \in \mathbb{R}^{p_k \times r_k},\ k = 1, \ldots, K$ are called factor matrices. Based on the decomposition, we obtain the Tucker tensor quantile regression model

$$Q_\tau(y|\boldsymbol{\mathcal{X}}) = \langle \boldsymbol{\mathcal{G}} \times_1 \boldsymbol{U}_1 \times_2 \boldsymbol{U}_2 \cdots \times_K \boldsymbol{U}_K, \boldsymbol{\mathcal{X}} \rangle. \tag{1}$$

As a result of decomposition, the number of parameters in model (1) reduces to $\prod_{k=1}^K r_k + \sum_{k=1}^K (p_k - r_k) r_k$ according to Zhang (2019), which is substantially smaller than $\prod_{k=1}^K p_k$.

Besides the Tucker decomposition, the CP decomposition and Tensor-Train (TT) decomposition (Oseledets, 2011) are also frequently used, see for example Guo et al. (2012); Zhou et al. (2013); Liu et al. (2020). In general it is hard to compare the dimension reduction ability of CP decomposition and Tucker decomposition. We choose Tucker decomposition partially due to that it is unique under mild assumptions and given the tensor the multilinear rank is easy to find while it is an NP hard problem to determine the CP rank (Håstad, 1990). TT decomposition is an efficient way to tackle tensor with large tensor order K. Furthermore, it may result in different TT format after permuting the modes of a tensor which may cause additional difficulty. For specificity, we only consider Tucker decomposition in the current work, while we acknowledge that investigations of other decompositions may also lead to fruitful results.

To estimate the coefficient of quantile regression model, Koenker and Bassett (1978) proposed to replace the ℓ_2 loss in mean regression by the check loss function, defined by

$$\rho_\tau(u) = \begin{cases} \tau u, & \text{if } u \geq 0, \\ (\tau - 1)u, & \text{if } u < 0, \end{cases}$$

or equivalently $\rho_\tau(u) = u(\tau - I(u \leq 0))$ where $I(.)$ is the indicator function that takes value 1 if the statement inside the bracket is true and 0 otherwise. For a given quantile τ, the estimate of model (1) with ranks $(r_1, r_2, \ldots, r_K)$ is obtained by minimizing $\sum_{i=1}^n \rho_\tau \left(y_i - \langle \boldsymbol{\mathcal{G}} \times_1 \boldsymbol{U}_1 \times_2 \boldsymbol{U}_2 \cdots \times_K \boldsymbol{U}_K, \boldsymbol{\mathcal{X}}_i \rangle\right)$. Thus the estimator of $\boldsymbol{\mathcal{A}}$ can be defined as

$$\begin{aligned} \widehat{\boldsymbol{\mathcal{A}}} =& \widehat{\boldsymbol{\mathcal{G}}} \times_1 \widehat{\boldsymbol{U}}_1 \times_2 \widehat{\boldsymbol{U}}_2 \cdots \times_K \widehat{\boldsymbol{U}}_K \\ \in& \underset{\boldsymbol{\mathcal{G}}, \boldsymbol{U}_k, k=1,\ldots,K}{\arg\min} \sum_{i=1}^n \rho_\tau \left(y_i - \langle \boldsymbol{\mathcal{G}} \times_1 \boldsymbol{U}_1 \times_2 \boldsymbol{U}_2 \cdots \times_K \boldsymbol{U}_K, \boldsymbol{\mathcal{X}}_i \rangle\right). \end{aligned} \tag{2}$$

Therefore, the problem of estimating $\boldsymbol{\mathcal{A}}$ is transformed to estimating the core tensor $\boldsymbol{\mathcal{G}}$ and factor matrices $\boldsymbol{U}_k,\ k = 1, \ldots, K$. Note that the core tensor and factor matrices are not identified since the decomposition is not unique (unless we put more stringent conditions on the tensor). Therefore, the minimizer is not unique. Our theoretical results hold for any minimizer in the set of minimizers.

By the fact that $\boldsymbol{\mathcal{A}}_{(k)} = \boldsymbol{U}_k \boldsymbol{\mathcal{G}}_{(k)} (\boldsymbol{U}_K \otimes \cdots \otimes \boldsymbol{U}_{k+1} \otimes \boldsymbol{U}_{k-1} \otimes \cdots \otimes \boldsymbol{U}_1)^T$, we can rewrite $\langle \boldsymbol{\mathcal{G}} \times_1 \boldsymbol{U}_1 \times_2 \boldsymbol{U}_2 \cdots \times_K \boldsymbol{U}_K, \boldsymbol{\mathcal{X}} \rangle$ as

$$\mathrm{vec}(\boldsymbol{\mathcal{X}})^T (\boldsymbol{U}_K \otimes \boldsymbol{U}_{K-1} \otimes \cdots \otimes \boldsymbol{U}_1) \mathrm{vec}(\boldsymbol{\mathcal{G}}) \tag{3}$$

and

$$\mathrm{vec}(\boldsymbol{\mathcal{X}}_{(k)})^T \left(\left[(\boldsymbol{U}_K \otimes \cdots \otimes \boldsymbol{U}_{k+1} \otimes \boldsymbol{U}_{k-1} \otimes \cdots \otimes \boldsymbol{U}_1) \boldsymbol{\mathcal{G}}_{(k)}^T \right] \otimes \boldsymbol{I}_{p_k} \right) \mathrm{vec}(\boldsymbol{U}_k), \tag{4}$$

where $\boldsymbol{I}_{p_k}$ is the $p_k \times p_k$ identity matrix. This implies that the objective function (2) is linear with respect to $\mathrm{vec}(\boldsymbol{\mathcal{G}})$ and $\mathrm{vec}(\boldsymbol{U}_k),\ k = 1, \ldots, K$, when the others are fixed. Equations (3) and (4) involve computation of Kronecker products which results in large and unwieldy matrices when the dimension is high. (3) and (4) are used mainly due to their elegant mathematical presentation. In practice, one can organize the computation in other ways. For example, for $K = 3$, since

$$\langle \boldsymbol{\mathcal{G}} \times_1 \mathbf{U}_1 \times_2 \mathbf{U}_2 \times_3 \mathbf{U}_3, \boldsymbol{\mathcal{X}} \rangle = \sum_{d_1=1}^{p_1} \sum_{d_2=1}^{p_2} \sum_{d_3=1}^{p_3} \sum_{s_1=1}^{r_1} \sum_{s_2=1}^{r_2} \sum_{s_3=1}^{r_3} x_{d_1 d_2 d_3} u_{1,d_1 s_1} u_{2,d_2 s_2} u_{3,d_3 s_3} g_{s_1 s_2 s_3},$$

where $x_{d_1 d_2 d_3}, u_{1,d_1 s_1}, u_{2,d_2 s_2}, u_{3,d_3 s_3}, g_{s_1 s_2 s_3}$ denotes the entries of $\boldsymbol{\mathcal{X}}, \mathbf{U}_1, \mathbf{U}_2, \mathbf{U}_3, \boldsymbol{\mathcal{G}}$, respectively, then it is easy to see that $\mathrm{vec}(\boldsymbol{\mathcal{X}})^T (\boldsymbol{U}_3 \otimes \boldsymbol{U}_2 \otimes \boldsymbol{U}_1)$ in (3) can be computed via the vectorization of $\boldsymbol{\mathcal{X}} \times_1 \mathbf{U}_1^T \times_2 \mathbf{U}_2^T \times_3 \mathbf{U}_3^T$, for example, to avoid directly computing the Kronecker product. Similarly, (4) can be dealt with using appropriate matrix products.

The following alternating update algorithm is used to find $\widehat{\boldsymbol{\mathcal{A}}}$. We note that we do not require $\boldsymbol{\mathcal{G}}, \boldsymbol{U}_k$ to be identified since we are only interested in the tensor $\boldsymbol{\mathcal{A}}$. In particular, in this part, we do not require $\boldsymbol{U}_k$ to be orthogonal which makes the optimization problem simpler. Such a choice for similar low-rank matrix/tensor models are often adopted in the literature (Liu et al., 2013; Udell et al., 2016). The alternating update algorithm obviously can decrease the objective function value in each step and the function value is guaranteed to converge.

Algorithm 1 Alternating update algorithm for low dimensional quantile regression

```
Initialized: $\boldsymbol{\mathcal{A}}^{(0)} = \arg\min \sum_{i=1}^n \rho_\tau(y_i - \mathrm{vec}(\boldsymbol{\mathcal{X}}_i)^T \mathrm{vec}(\boldsymbol{\mathcal{A}}))$
Perform HOSVD: $\boldsymbol{\mathcal{A}}^{(0)} = \boldsymbol{\mathcal{G}}^{(0)} \times_1 \boldsymbol{U}_1^{(0)} \times_2 \boldsymbol{U}_2^{(0)} \cdots \times_K \boldsymbol{U}_K^{(0)}$ with predetermined ranks $(r_1, \dots, r_K)$
**Repeat** $\ell = 0, 1, 2, \dots$
    For $k = 1, \dots, K$
        $\boldsymbol{U}_k^{(\ell+1)} = \arg\min_{\boldsymbol{U}_k} \sum_{i=1}^n \rho_\tau\Big(y_i - \mathrm{vec}(\boldsymbol{\mathcal{X}}_{i,(k)})^T \Big(\big[(\boldsymbol{U}_K^{(\ell)} \otimes \cdots \otimes \boldsymbol{U}_{k+1}^{(\ell)} \otimes \boldsymbol{U}_{k-1}^{(\ell+1)} \otimes \cdots \otimes \boldsymbol{U}_1^{(\ell+1)}) \boldsymbol{\mathcal{G}}_{(k)}^{(\ell+1)T}\big] \otimes \boldsymbol{I}_{p_k}\Big) \mathrm{vec}(\boldsymbol{U}_k)\Big)$
    End for
    $\boldsymbol{\mathcal{G}}^{(\ell+1)} = \arg\min_{\boldsymbol{\mathcal{G}}} \sum_{i=1}^n \rho_\tau\Big(y_i - \mathrm{vec}(\boldsymbol{\mathcal{X}}_{i,(1)})^T (\boldsymbol{U}_K^{(\ell+1)} \otimes \boldsymbol{U}_{K-1}^{(\ell+1)} \otimes \cdots \otimes \boldsymbol{U}_1^{(\ell+1)}) \mathrm{vec}(\boldsymbol{\mathcal{G}}_{(1)})\Big)$
    $\boldsymbol{\mathcal{A}}^{(\ell+1)} = \boldsymbol{\mathcal{G}}^{(\ell+1)} \times_1 \boldsymbol{U}_1^{(\ell+1)} \times_2 \boldsymbol{U}_2^{(\ell+1)} \cdots \times_K \boldsymbol{U}_K^{(\ell+1)}$
**Until convergence**
```

Remark 1 *Note that we omit the intercept in most parts of the paper for simplicity of exposition. When an intercept is added, its estimation can be combined with the $\boldsymbol{\mathcal{G}}$-update step. In this way, the $\boldsymbol{\mathcal{G}}$-update step becomes $(\mu^{(\ell+1)}, \boldsymbol{\mathcal{G}}^{(\ell+1)}) = \arg\min_{\mu, \boldsymbol{\mathcal{G}}} \sum_{i=1}^n \rho_\tau\Big(y_i - \mu - vec(\boldsymbol{\mathcal{X}}_{i,(1)})^T (\boldsymbol{U}_K^{(\ell)} \otimes \boldsymbol{U}_{K-1}^{(\ell)} \otimes \cdots \otimes \boldsymbol{U}_1^{(\ell)}) vec(\boldsymbol{\mathcal{G}}_{(1)})\Big)$, and the y_i in $\boldsymbol{U}$-update step is replaced by $y_i - \mu^\ell$. The same modification is used in the sparse case in the following section.*

2.3 Sparse Tucker tensor quantile regression

When the number of variables $p_1 p_2 \cdots p_K$ is large, it is desirable to incorporate sparsity. In our case study later, sparsity is mainly used for dimension reduction without much interpretability.

Assume that there exists a Tucker decomposition for $\boldsymbol{\mathcal{A}}$ such that the factor matrices $\boldsymbol{U}_1, \dots, \boldsymbol{U}_K$ are sparse orthogonal matrices. We impose ℓ_1 penalty for variable selection and hence the estimator is

$$\begin{aligned} \widehat{\boldsymbol{\mathcal{A}}} =& \widehat{\boldsymbol{\mathcal{G}}} \times_1 \widehat{\boldsymbol{U}}_1 \times_2 \widehat{\boldsymbol{U}}_2 \cdots \times_K \widehat{\boldsymbol{U}}_K \\ =& \arg\min \frac{1}{n} \sum_{i=1}^n \rho_\tau\Big(y_i - \langle \boldsymbol{\mathcal{G}} \times_1 \boldsymbol{U}_1 \times_2 \boldsymbol{U}_2 \cdots \times_K \boldsymbol{U}_K, \boldsymbol{\mathcal{X}}_i \rangle\Big) + \lambda \|\boldsymbol{U}_K \otimes \cdots \otimes \boldsymbol{U}_1\|_1, \end{aligned} \tag{5}$$

subject to $\boldsymbol{U}_k^T \boldsymbol{U}_k = \boldsymbol{I}_{r_k}, \ k = 1, \dots, K$.

By the definition of the ℓ_1 norm, we have

$$\begin{aligned} & \|\boldsymbol{U}_K \otimes \cdots \otimes \boldsymbol{U}_1\|_1 \\ = & \sum_{i_K=1}^{p_K} \sum_{i_{K-1}=1}^{p_{K-1}} \cdots \sum_{i_1=1}^{p_1} \sum_{j_K=1}^{r_K} \sum_{j_{K-1}=1}^{r_{K-1}} \cdots \sum_{j_1}^{r_1} |u_{K,i_K j_K} u_{K-1,i_{K-1} j_{K-1}} \cdots u_{1,i_1 j_1}| \\ = & \left(\sum_{i_1=1}^{p_1} \sum_{j_1=1}^{r_1} |u_{1,i_1 j_1}|\right) \times \cdots \times \left(\sum_{i_K=1}^{p_K} \sum_{j_K=1}^{r_K} |u_{K,i_K j_K}|\right) \\ = & \|\boldsymbol{U}_1\|_1 \|\boldsymbol{U}_2\|_1 \cdots \|\boldsymbol{U}_K\|_1. \end{aligned}$$

Algorithm 2 Alternating update algorithm for high dimensional quantile regression

```
Initialized: $\boldsymbol{\mathcal{A}}^{(0)} = \arg\min \sum_{i=1}^n \rho_\tau(y_i - \mathrm{vec}(\boldsymbol{\mathcal{X}}_i)^T \mathrm{vec}(\boldsymbol{\mathcal{A}}))$
Perform HOSVD: $\boldsymbol{\mathcal{A}}^{(0)} = \boldsymbol{\mathcal{G}}^{(0)} \times_1 \boldsymbol{U}_1^{(0)} \times_2 \boldsymbol{U}_2^{(0)} \cdots \times_K \boldsymbol{U}_K^{(0)}$ with predetermined ranks $(r_1, \dots, r_K)$
Repeat $\ell = 0, 1, 2, \dots$
  For $k = 1, \dots, K$
    $\boldsymbol{U}_k^{(\ell+1)} = \arg\min_{\boldsymbol{U}_k^T \boldsymbol{U}_k = \boldsymbol{I}_{r_k}} \frac{1}{n} \sum_{i=1}^n \rho_\tau \Big( y_i - \mathrm{vec}(\boldsymbol{\mathcal{X}}_{i,(k)})^T \Big( \big[ (\boldsymbol{U}_K^{(\ell)} \otimes \cdots \otimes \boldsymbol{U}_{k+1}^{(\ell)} \otimes \boldsymbol{U}_{k-1}^{(\ell+1)} \otimes \cdots \otimes \boldsymbol{U}_1^{(\ell+1)})$
        $\boldsymbol{\mathcal{G}}_{(k)}^{(\ell+1)T} \big] \otimes \boldsymbol{I}_{p_k} \Big) \mathrm{vec}(\boldsymbol{U}_k) \Big) + \lambda \|\boldsymbol{U}_1^{(\ell+1)}\|_1 \cdots \|\boldsymbol{U}_{k-1}^{(\ell+1)}\|_1 \|\boldsymbol{U}_{k+1}^{(\ell)}\|_1 \cdots \|\boldsymbol{U}_K^{(\ell)}\|_1 \|\boldsymbol{U}_k\|_1$
  End for
  $\boldsymbol{\mathcal{G}}^{(\ell+1)} = \arg\min_{\boldsymbol{\mathcal{G}}} \frac{1}{n} \sum_{i=1}^n \rho_\tau \Big( y_i - \mathrm{vec}(\boldsymbol{\mathcal{X}}_i)^T (\boldsymbol{U}_K^{(\ell+1)} \otimes \boldsymbol{U}_{K-1}^{(\ell+1)} \otimes \cdots \otimes \boldsymbol{U}_1^{(\ell+1)}) \mathrm{vec}(\boldsymbol{\mathcal{G}}) \Big)$
  $\boldsymbol{\mathcal{A}}^{(\ell+1)} = \boldsymbol{\mathcal{G}}^{(\ell+1)} \times_1 \boldsymbol{U}_1^{(\ell+1)} \times_2 \boldsymbol{U}_2^{(\ell+1)} \cdots \times_K \boldsymbol{U}_K^{(\ell+1)}$
Until convergence
```

Therefore, the ℓ_1 penalty is imposed jointly on the ℓ_1 norm of all the factor matrices. Several remarks are in order. First, we do not penalize the core tensor matrix since usually r_1, r_2, r_3 are small. If one really wants, one can also add a penalty on $\boldsymbol{\mathcal{G}}$. Entries of $\boldsymbol{\mathcal{G}}$ represents interactions between latent dimensions and selection of nonzero entries may be of interest in some cases. However, in our application, we simply use sparsity as a way to reduce dimension and the size of $\boldsymbol{\mathcal{G}}$ is already very small and thus we do not impose sparsity on the core tensor. Second, putting a penalty on the product $\|\boldsymbol{U}_K \otimes \cdots \otimes \boldsymbol{U}_1\|_1 = \|\boldsymbol{U}_1\|_1 \|\boldsymbol{U}_2\|_1 \cdots \|\boldsymbol{U}_K\|_1$ is somewhat unusual, but is motivated by the form (3) in which the product of factor matrices appears. Using such a penalty makes it much more convenient, to say the least, to perform our theoretical analysis later. In practice, we find it performs very similar to using the more conventional penalty $\lambda(\|\boldsymbol{U}_1\|_1 + \|\boldsymbol{U}_2\|_1 + \cdots + \|\boldsymbol{U}_K\|_1)$. Third, we only use one tuning parameter λ, which is only suitable if all factor matrices are believed to have similar degrees of sparsity. For greater flexibility, one should instead use $\lambda_1\|\boldsymbol{U}_1\|_1 + \lambda_2\|\boldsymbol{U}_2\|_1 + \cdots + \lambda_K\|\boldsymbol{U}_K\|_1$. In our current numerical examples, we find there is no need to use the more flexible penalty which leads to the computational burden of having to tune multiple parameters. Finally, unlike the non-sparse case, here we would require the orthogonality constraint. The reason is that without the constraint, due to scale indeterminacy (multiplying $\boldsymbol{\mathcal{G}}$ by any constant while dividing the same constant on $\mathbf{U}_k$ will result in the same $\boldsymbol{\mathcal{A}}$), the penalty will make all $\mathbf{U}_k$ arbitrarily close to zero while making $\boldsymbol{\mathcal{G}}$ very large. This is in stark contrast with Algorithm 1, where we do not need to use any constraints when there is no penalty and the algorithm is much simpler there. Alternatively, a ridge penalty on $\boldsymbol{\mathcal{G}}$ can also be used with an additional tuning parameter. However, in the high-dimensional case, when there is no orthogonal constraint, we often find such estimators are more unstable with larger variances.

The estimate $\widehat{\boldsymbol{\mathcal{A}}}$ can be computed by an alternating update algorithm similar to that in Section 2.2, see Algorithm 2. However, the update of $\boldsymbol{U}_k, k = 1, \dots, K$, is a nonconvex optimization problem due to the orthogonality constraint, thus it can not be solved by ordinary quantile regression as in Section 2.2. In order to separate the orthogonality constraint and ℓ_1 regularization, we propose to use an ADMM algorithm motived by Yu et al. (2017).

More specifically, the update of $\boldsymbol{U}_k, k = 1, \dots, K$ takes the common form

$$\min_{\boldsymbol{B}} \frac{1}{n} \rho_\tau\big(\boldsymbol{y} - \boldsymbol{Z}\text{vec}(\boldsymbol{B})\big) + \lambda \|\boldsymbol{B}\|_1,\ s.t.\ \boldsymbol{B}^T\boldsymbol{B} = \boldsymbol{I}, \tag{6}$$

where $\rho_\tau\big(\boldsymbol{y} - \boldsymbol{Z}\text{vec}(\boldsymbol{B})\big) = \sum_{i=1}^n \rho_\tau\big(y_i - \boldsymbol{z}_i\text{vec}(\boldsymbol{B})\big)$. Rewriting (6) in an equivalent form by introducing auxiliary variables $\boldsymbol{\beta}_b$, $\boldsymbol{r}$ and $\boldsymbol{P}$, we have

$$\begin{aligned} &\arg\min_{\boldsymbol{r},\boldsymbol{\beta}_b,\boldsymbol{\beta},\boldsymbol{P}} \frac{1}{n}\rho_\tau(\boldsymbol{r}) + \lambda\|\boldsymbol{\beta}\|_1 \\ &\text{subject to } \boldsymbol{y} - \boldsymbol{Z}\boldsymbol{\beta}_b = \boldsymbol{r},\ \boldsymbol{\beta}_b = \boldsymbol{\beta}, \\ &\boldsymbol{B}_b = \boldsymbol{P},\ \boldsymbol{P}^T\boldsymbol{P} = \boldsymbol{I}, \end{aligned} \tag{7}$$

where $\boldsymbol{\beta} = \text{vec}(\boldsymbol{B})$ and $\boldsymbol{\beta}_b = \text{vec}(\boldsymbol{B}_b)$. Using augmented Lagrangian, the ADMM algorithm consists of the following steps,

$$\begin{aligned}
\boldsymbol{\beta}^{(j+1)} &= \arg\min_{\boldsymbol{\beta}} \lambda\|\boldsymbol{\beta}\|_1 + \frac{\gamma}{2}\|\boldsymbol{\beta}_b^{(j)} - \boldsymbol{\beta} + \frac{1}{\gamma}\boldsymbol{\eta}^{(j)}\|^2 \\
\boldsymbol{r}^{(j+1)} &= \arg\min_{\boldsymbol{r}} \frac{1}{n}\rho_\tau(\boldsymbol{r}) + \frac{\gamma}{2}\|\boldsymbol{y} - \boldsymbol{Z}\boldsymbol{\beta}_b^{(j)} - \boldsymbol{r} + \frac{1}{\gamma}\boldsymbol{u}^{(j)}\|^2 \\
\boldsymbol{P}^{(j+1)} &= \arg\min_{\boldsymbol{P}} \frac{\gamma}{2}\|\boldsymbol{B}_b^{(j)} - \boldsymbol{P} + \frac{1}{\gamma}\boldsymbol{E}^{(j)}\|_F^2, \quad \text{subject to } \boldsymbol{P}^T\boldsymbol{P} = \boldsymbol{I} \\
\boldsymbol{\beta}_b^{(j+1)} &= \arg\min_{\boldsymbol{\beta}_b} \frac{\gamma}{2}\|\boldsymbol{y} - \boldsymbol{Z}\boldsymbol{\beta}_b - \boldsymbol{r}^{(j+1)} + \frac{1}{\gamma}\boldsymbol{u}^{(j)}\|^2 + \frac{\gamma}{2}\|\boldsymbol{\beta}_b - \boldsymbol{\beta}^{(j+1)} + \frac{1}{\gamma}\boldsymbol{\eta}^{(j)}\|^2 \\
&\quad + \frac{\gamma}{2}\|\boldsymbol{B}_b - \boldsymbol{P}^{(j+1)} + \frac{1}{\gamma}\boldsymbol{E}^{(j)}\|_F^2 \\
\boldsymbol{E}^{(j+1)} &= \boldsymbol{E}^{(j)} + \gamma(\boldsymbol{B}_b^{(j+1)} - \boldsymbol{P}^{(j+1)}) \\
\boldsymbol{u}^{(j+1)} &= \boldsymbol{u}^{(j)} + \gamma(\boldsymbol{y} - \boldsymbol{Z}\boldsymbol{\beta}_b^{(j+1)} - \boldsymbol{r}^{(j+1)}) \\
\boldsymbol{\eta}^{(j+1)} &= \boldsymbol{\eta}^{(j)} + \gamma(\boldsymbol{\beta}_b^{(j+1)} - \boldsymbol{\beta}^{(j+1)}).
\end{aligned}$$

All updates for $\boldsymbol{\beta}$, $\boldsymbol{\beta}_b$, $\boldsymbol{r}$, $\mathbf{P}$ above can be obtained in closed form, as detailed in Algorithm 3. We note that for general nonconvex problems the ADMM algorithm does not have satisfying theoretical convergence guarantee, but in our numerical studies we observe good convergence behavior for the algorithm although the results are not presented here. In particular, we always observe that $\boldsymbol{B}_b$ and $\boldsymbol{B}$ are very close to being orthogonal.

3. Theoretical properties

In this section, we investigate the statistical properties of the proposed quantile tensor regression method. In particular, we establish the asymptotic normality for the estimate of nonsparse quantile tensor regression under mild general conditions, in particular showing that the estimator is $\sqrt{n}$-consistent. Then we derive an upper bound for the estimation error under the sparse setting, which shows that the estimate converges to the true value at a rate depending on sample size, number of nonzero parameters and logarithm of dimension

Algorithm 3 ADMM for sparse and orthogonal regression

```
Initialize: $\boldsymbol{B}_b^{(0)} = \boldsymbol{U}_k^{(\ell)}$, $\boldsymbol{P}^{(0)} = \boldsymbol{U}_k^{(\ell)}$, $\boldsymbol{E}^{(0)} = \mathbf{0}$, $\boldsymbol{\eta}^{(0)} = \boldsymbol{u}_b^{(0)} = \mathbf{0}$
Repeat $j = 0, 1, 2, \ldots$
  $\boldsymbol{\beta}^{(j+1)} = [\boldsymbol{\beta}_b^{(j)} + \boldsymbol{\eta}^{(j)}/\gamma - \lambda/\gamma]_+ - [-\boldsymbol{\beta}_b^{(j)} - \boldsymbol{\eta}^{(j)}/\gamma - \lambda/\gamma]_+$
  $\boldsymbol{r}^{(j+1)} = [\boldsymbol{y} - \boldsymbol{Z}\boldsymbol{\beta}_b^{(j)} + \boldsymbol{u}^{(j)}/\gamma - \frac{\tau}{n\gamma}\mathbf{1}_n]_+ - [-\boldsymbol{y} + \boldsymbol{Z}\boldsymbol{\beta}_b^{(j)} - \boldsymbol{u}^{(j)}/\gamma + \frac{\tau-1}{n\gamma}\mathbf{1}_n]_+$
  Compute SVD $\boldsymbol{B}_b^{(j)} + \boldsymbol{E}^{(j)}/\gamma = \boldsymbol{V}_1\boldsymbol{D}\boldsymbol{V}_2^T$
  $\boldsymbol{P}^{(j+1)} = \boldsymbol{V}_1\boldsymbol{V}_2^T$
  $\boldsymbol{\beta}_b^{(j+1)} = (\boldsymbol{Z}^T\boldsymbol{Z} + 2\boldsymbol{I})^{-1}\big[\boldsymbol{Z}^T(\boldsymbol{y} - \boldsymbol{r}^{(j+1)} + \boldsymbol{u}^{(j)}/\gamma) - \boldsymbol{\eta}^{(j)}/\gamma + \boldsymbol{\beta}^{(j+1)} + \mathrm{vec}(\boldsymbol{P}^{(j+1)}) - \mathrm{vec}(\boldsymbol{E}^{(j)})/\gamma\big]$
  $\boldsymbol{E}^{(j+1)} = \boldsymbol{E}^{(j)} + \gamma(\boldsymbol{B}_b^{(j+1)} - \boldsymbol{P}^{(j+1)})$
  $\boldsymbol{u}^{(j+1)} = \boldsymbol{u}^{(j)} + \gamma(\boldsymbol{y} - \boldsymbol{Z}\boldsymbol{\beta}_b^{(j+1)} - \boldsymbol{r}^{(j+1)})$
  $\boldsymbol{\eta}^{(j+1)} = \boldsymbol{\eta}^{(j)} + \gamma(\boldsymbol{\beta}_b^{(j+1)} - \boldsymbol{\beta}^{(j+1)})$
Until convergence
```

p. The logarithmic dependence on p means the procedure can be applied to very large tensors, although limited by computational efficiency in implementation.

We first consider the asymptotic distribution of the estimator for the nonsparse Tucker tensor quantile regression discussed in Section 2.2, assuming the dimensions $p_1, \ldots, p_K$ are fixed. Let $\boldsymbol{\phi} = (\mathrm{vec}(\boldsymbol{\mathcal{G}})^T, \mathrm{vec}(\boldsymbol{U}_1)^T, \ldots, \mathrm{vec}(\boldsymbol{U}_K)^T)^T$ be the true parameters (the "true" $\boldsymbol{\phi}$ is not unique but any that corresponds to the true $\boldsymbol{\mathcal{A}}$ will do). Let $\boldsymbol{h}(\boldsymbol{\phi}) = \mathrm{vec}(\boldsymbol{\mathcal{A}}) = \mathrm{vec}\big(\boldsymbol{\mathcal{G}}\times_1\boldsymbol{U}_1\times_2\boldsymbol{U}_2\cdots\times_K\boldsymbol{U}_K\big)$ considered as a function of $\boldsymbol{\phi}$. Define the $\prod_{k=1}^K p_k \times \big(\prod_{k=1}^K r_k + \sum_{k=1}^K p_k r_k\big)$ Jacobian matrix $\boldsymbol{H}$ as

$$\begin{aligned}\boldsymbol{H} = \frac{\partial \boldsymbol{h}}{\partial \boldsymbol{\phi}} = \Big(&\boldsymbol{U}_K \otimes \cdots \otimes \boldsymbol{U}_1, \big[(\boldsymbol{U}_K \otimes \cdots \otimes \boldsymbol{U}_2)\boldsymbol{\mathcal{G}}_{(1)}^T\big] \otimes \boldsymbol{I}_{p_1},\\ &\boldsymbol{T}_{21}\{\big[(\boldsymbol{U}_K \otimes \cdots \otimes \boldsymbol{U}_3 \otimes \boldsymbol{U}_1)\boldsymbol{\mathcal{G}}_{(2)}^T\big] \otimes \mathbf{I}_{p_2}\}, \ldots,\\ &\boldsymbol{T}_{K1}\{\big[(\boldsymbol{U}_{K-1} \otimes \cdots \otimes \boldsymbol{U}_1)\boldsymbol{\mathcal{G}}_{(K)}^T\big] \otimes \mathbf{I}_{p_K}\}\Big),\end{aligned}$$

where $\boldsymbol{T}_{ij}$ is the permutation matrix such that $\mathrm{vec}(\boldsymbol{\mathcal{A}}_{(j)}) = \boldsymbol{T}_{ij}\mathrm{vec}(\boldsymbol{\mathcal{A}}_{(i)})$. Let $F_{\varepsilon|\boldsymbol{x}}(t)$ and $f_{\varepsilon|\boldsymbol{x}}(t)$ be the conditional distribution function and conditional density function of random errors $\varepsilon := y - \langle \boldsymbol{\mathcal{X}}, \boldsymbol{\mathcal{A}} \rangle$. In addition, let $\boldsymbol{x}_i = \mathrm{vec}(\boldsymbol{\mathcal{X}}_{i,(1)})$, $\boldsymbol{D}_0 = E(\boldsymbol{x}_i\boldsymbol{x}_i^T)$, $\boldsymbol{D}_1 = E(\boldsymbol{x}_i\boldsymbol{x}_i^T f_{\varepsilon|\boldsymbol{x}}(0))$ and denote the smallest eigenvalue of a symmetric matrix $\boldsymbol{D}$ by $\delta_{\min}(\boldsymbol{D})$. Finally we define $\boldsymbol{\Gamma} = \tau(1-\tau)\boldsymbol{D}_1^{-1}\boldsymbol{D}_0\boldsymbol{D}_1^{-1}$. In order to establish the asymptotic properties of the estimator (2), we assume the following conditions.

C1. $f_{\varepsilon|\boldsymbol{x}}(0)$ is bounded away from zero uniformly over the support of $\mathbf{x}$, and both $f_{\varepsilon|\boldsymbol{x}}(.)$ and its derivative are uniformly bounded.

C2. $E\|\boldsymbol{x}\|^3 < \infty$. The matrix $\boldsymbol{D}_0$ is positive definite.

C3. The parameter space for $\boldsymbol{h}$ is bounded.

Conditions C1 and C2 are mild regularity assumptions commonly used in quantile regression models (Wang et al., 2009; Belloni and Chernozhukov, 2011). Condition C1 imposes smoothness assumptions on the conditional density of random errors, which is satisfied by most distributions such as Gaussian and exponential distribution. Condition C2 is an assumption on existence of moments to ensure asymptotic normality of the estimator. It is trivially satisfied if $\boldsymbol{x}$ is bounded. The boundedness of parameter space in Condition C3 is often necessary in nonconvex models in order to apply empirical process theory. In practice, one usually searches within a large but bounded region and thus this assumption is not overly stringent. We establish the asymptotic distribution of $\boldsymbol{h}(\widehat{\boldsymbol{\phi}})$ in the following theorem, with proofs relegated to the appendix.

Theorem 1 *Suppose conditions C1-C3 hold, then as $n \to \infty$,*

$$\sqrt{n}\big(\boldsymbol{h}(\widehat{\boldsymbol{\phi}}) - \boldsymbol{h}(\boldsymbol{\phi})\big) \xrightarrow{d} N(0, \boldsymbol{\Sigma}), \tag{8}$$

where $\boldsymbol{\Sigma} = \boldsymbol{P}\boldsymbol{\Gamma}\boldsymbol{P}^T$, and $\boldsymbol{P} = \boldsymbol{H}(\boldsymbol{H}^T\boldsymbol{D}_1\boldsymbol{H})^{-}\boldsymbol{H}^T\boldsymbol{D}_1$. Here $(.)^{-}$ denotes the pseudo-inverse.

Note that $\boldsymbol{\Gamma}$ is the asymptotic variance of conventional quantile regression. This theorem in particular shows that the estimator is $\sqrt{n}$-consistent. Such asymptotic results can potentially lead to interval estimates which however is not our focus in this paper (we are mainly interested in its prediction performance especially in high dimensions).

Next we derive the error bound for the estimation error of sparse Tucker tensor quantile regression (5) in high dimensions. Let $r = \prod_{k=1}^K r_k$, $p = \prod_{k=1}^K p_k$, $\boldsymbol{U} = \boldsymbol{U}_K \otimes \cdots \otimes \boldsymbol{U}_1$ and $\boldsymbol{g} = \mathrm{vec}(\boldsymbol{\mathcal{G}})$. To obtain the upper bound of estimation error, we assume the following conditions.

C4. The factor matrices are sparse and satisfy $\|\boldsymbol{U}_k\|_0 \le s_k$ for $1 \le k \le K$.

C5. The parameter space for $\boldsymbol{\mathcal{G}}$ is $\boldsymbol{\Omega}_{\mathbf{g}} = \{\boldsymbol{\mathcal{G}} : \|\mathrm{vec}(\boldsymbol{\mathcal{G}})\|_\infty \le \bar{g} < \infty\}$ for some $\bar{g} > 0$. The parameter space for $\mathbf{U}_k$ is $\boldsymbol{\Omega}_k = \{\mathbf{U}_k : \boldsymbol{U}_k \in \mathbb{R}^{p_k \times r_k}, \boldsymbol{U}_k^T\boldsymbol{U}_k = \boldsymbol{I}\}$. We let $\boldsymbol{\Omega}_{\mathbf{U}} = \{\mathbf{U} = \mathbf{U}_K \otimes \cdots \otimes \mathbf{U}_1 \in \mathbb{R}^{p \times r} : \mathbf{U}_k \in \boldsymbol{\Omega}_k\}$.

C6. $\mathbf{x} = \mathrm{vec}(\boldsymbol{\mathcal{X}})$ is sub-Gaussian in the sense that $E\exp\{\mathbf{a}^T\mathbf{x}\} \le C\exp\{C\|\mathbf{a}\|^2\}$ for any vector $\mathbf{a}$ where $C > 0$ is a constant. The eigenvalues of $\boldsymbol{D}_0$ are bounded away from zero and infinity.

C7. The maximum and the minimum non-zero singular values of the rank-r_k matrix $\boldsymbol{\mathcal{A}}_{(k)}$ is given by $\sigma_{max,k}$ and $\sigma_{min,k}$ respectively.

C8. Let $\boldsymbol{\Omega} = \{\boldsymbol{\Delta} \in \mathbb{R}^p : \boldsymbol{\Delta} = (\mathbf{U} + \boldsymbol{\Delta}_{\mathbf{U}})(\mathbf{g} + \boldsymbol{\Delta}_{\mathbf{g}}) - \mathbf{U}\mathbf{g}, \mathbf{U} + \boldsymbol{\Delta}_{\mathbf{U}} \in \boldsymbol{\Omega}_{\mathbf{U}}, \mathbf{g} + \boldsymbol{\Delta}_{\mathbf{g}} \in \boldsymbol{\Omega}_{\mathbf{g}}, \|(\boldsymbol{\Delta}_{\boldsymbol{U}})_{S^c}\|_1 \le 3\|(\boldsymbol{\Delta}_{\boldsymbol{U}})_S\|_1 + \|\boldsymbol{\Delta}_{\mathbf{g}}\|_1\}$, where S is the indices of the nonzero entries of $\mathbf{U}$ with S^c its complement, $(\boldsymbol{\Delta}_{\mathbf{U}})_S$, for example, denotes the vector containing the entries of $\boldsymbol{\Delta}_{\mathbf{U}}$ indexed in S. We assume that there exists some positive constant c_1 such that for any $\boldsymbol{\Delta} \in \boldsymbol{\Omega}$ with $\|\boldsymbol{\Delta}\| = t$,

$$Q(\mathbf{U}\mathbf{g} + \boldsymbol{\Delta}) \ge c_1(t^2 \wedge t), \forall t > 0,$$

where $Q(\boldsymbol{\delta}) = E[\rho_\tau(y - \mathbf{x}^T\boldsymbol{\delta}) - \rho_\tau(y - \mathbf{x}^T\mathbf{U}\mathbf{g})]$.

Condition C4 constrains the sparsity of factor matrices. The condition $\|\mathbf{U}_k\|_0 \le s_k$ implies $\|\mathbf{U}\|_0 \le \prod_{k=1}^K s_k =: s$ where $\mathbf{U} = \mathbf{U}_K \otimes \cdots \otimes \mathbf{U}_1$. Note that we do not require an upper bound for s_k. Thus, the following Theorem 2 holds for any s, but the upper bound will increase as s goes up. The algorithm can always be applied whether the true matrix is sparse or not. Condition C5 restricts the parameter space for $\boldsymbol{\mathcal{G}}$ and $\boldsymbol{U}_k$, which is related to C3 in the fixed-dimensional case. We use the upper bound $\bar{g}$ for technical reasons in the proof. In the high-dimensional setting, we require a stronger distribution assumption on the predictors as in C6. It is satisfied if the components of $\mathbf{x}$ are bounded and independent, although sub-Gaussianity is much more general than boundedness. C7 assumes $\boldsymbol{\mathcal{A}}$ is low-rank. The bounds will be simplified if we assume $\sigma_{max,k}, \sigma_{min,k}$ are bounded away from zero and infinity. C8 is a high-level assumption on local strong convexity of the expected loss at the minimizer. Lemma 4 in Belloni and Chernozhukov (2011) showed that C8 holds if the quantity $\inf_{\boldsymbol{\Delta}\in\boldsymbol{\Omega},\boldsymbol{\Delta}\neq 0} \frac{(E|\boldsymbol{x}^T\boldsymbol{\Delta}|^2)^{3/2}}{E|\boldsymbol{x}^T\boldsymbol{\Delta}|^3}$ is bounded away from zero. This quantity is indeed bounded away from zero when, for example, $\mathbf{x}$ is Gaussian.

Theorem 2 *Suppose conditions C1 and C4-C8 hold. Assume* $\lambda \geq c_2\bar{g}\sqrt{\log(p\vee n)/n}$ *for some sufficiently large constant* c_2*, Then with probability* $1-(p\vee n)^{-c_3}$ *for some constant* $c_3 > 0$*,*

$$\|\widehat{\boldsymbol{\mathcal{A}}} - \boldsymbol{\mathcal{A}}\|_F \leq Cb_n\lambda/\bar{g}, \ \textit{for some constant } C > 0$$

if $d_n\lambda/\bar{g} = o(1)$*. Here* b_n, d_n *are defined as*

$$b_n = C\bar{g}\sqrt{s}\left(\sum_k \sqrt{\frac{r}{r_k}}\frac{\sigma_{max,k}}{\sigma^2_{min,k}}\right) + C(\bar{g}+1)\sqrt{r}\left(1+\sum_k \frac{\|\boldsymbol{\mathcal{A}}\|_F\sigma_{max,k}}{\sigma^2_{min,k}}\right),$$

and

$$d_n = C\bar{g}\sqrt{s}\left(\sum_k \sqrt{\frac{r}{r_k}}\frac{1}{\sigma^2_{min,k}}\right) + C(\bar{g}+1)\sqrt{r}\left(\sum_k \frac{\|\boldsymbol{\mathcal{A}}\|_F}{\sigma^2_{min,k}}\right).$$

The bound can be simplified significantly if we assume $\|\boldsymbol{\mathcal{A}}\|_F$, $\sigma_{max,k}$, $\bar{g}$, r are all bounded, and $\sigma_{min,k}$ are bounded away from zero. In this case, we obtain $\|\widehat{\boldsymbol{\mathcal{A}}}-\boldsymbol{\mathcal{A}}\|_F \leq C\sqrt{s\log(p\vee n)/n}$ when we set $\lambda \asymp \sqrt{\log(p\vee n)/n}$. This result gives the rate of convergence of the sparse quantile tensor regression estimator. The upper bound depends on the sample size n and the number of non-zero parameters s, and the dimension p only has a logarithmic effect. It also shows that the low-rank structure (smaller r_k) leads to better risk bounds.

4. Simulations

In this section, we carry out simulation studies to investigate the finite sample performances of the proposed methods.

4.1 Low-dimensional tensor quantile regression

We first consider the low-dimensional non-sparse estimator presented in Section 2.2. We examine the performances of the proposed algorithm under a variety of dimensions, sample sizes and signal strengths. Specifically, the response is generated by $y_i = \langle \boldsymbol{\mathcal{A}}, \boldsymbol{\mathcal{X}}_i\rangle + \varepsilon_i$, where $\boldsymbol{\mathcal{X}}_i$ is a $p_1 \times p_2 \times p_3$ tensor with standard normal entries, and random errors ε_i are generated independently from normal distribution $N(-q_\tau, \sigma^2)$ with q_τ being the τth quantile of $N(0,1)$. Coefficient $\boldsymbol{\mathcal{A}}$ is generated by $\boldsymbol{\mathcal{G}} \times_1 \boldsymbol{U}_1 \times_2 \boldsymbol{U}_2 \times_3 \boldsymbol{U}_3$. The entries of the core tensor $\boldsymbol{\mathcal{G}}$ are standard normal values. The factor matrix $\boldsymbol{U}_k$ is obtained from a QR decomposition of a $p_k \times r_k$ matrix with independent standard normal entries. We use dimensions $(p_1, p_2, p_3) = (5, 5, 5)$ and $(10, 10, 10)$. Rank (r_1, r_2, r_3) is chosen to be either $(2, 2, 2)$ or $(3, 3, 3)$. In addition, we choose the sample size $n = 1000$ and 2000, and noise level $\sigma = 0.5$ and 1.

To select the ranks in the proposed method, we use the following BIC

$$\mathrm{BIC} = \log\left(\frac{1}{n}\sum_{i=1}^{n} \rho_\tau(y_i - \langle \widehat{\boldsymbol{\mathcal{A}}}, \boldsymbol{\mathcal{X}} \rangle)\right) + \frac{df}{2n}\log n,$$

where the degrees of freedom (df) is defined as $df = r_1 r_2 \cdots r_K + \sum_{k=1}^{K} r_k(p_k - r_k)$. We compare the performances of the proposed method with lasso quantile regression (using the *hqreg* package in R, Yi and Huang, 2017). Table 1 presents the estimation errors (EE) measured by $\|\widehat{\boldsymbol{\mathcal{A}}} - \boldsymbol{\mathcal{A}}\|_F$, and the percentage that the true rank is selected, based on 200 replications. It can be seen that the estimation error of the proposed method is smaller than the lasso method in all settings (nothing unexpected here since the true model involves a low-rank tensor), and the percentage of choosing the true rank by BIC is always close to 1.

4.2 High-dimensional sparse tensor quantile regression

For this experiment, we take ranks $(r_1, r_2, r_3) = (2, 2, 2)$, and consider larger dimensions $(p_1, p_2, p_3) = (10, 10, 10)$, $(20, 20, 20)$ and $(100, 100, 3)$ and smaller sample sizes $n = 200, 400, 800$. $\boldsymbol{\mathcal{A}}$ is still generated by $\boldsymbol{\mathcal{G}} \times_1 \boldsymbol{U}_1 \times_2 \boldsymbol{U}_2 \times_3 \boldsymbol{U}_3$. A sparse factor matrix $\mathbf{U}$ is generated as follows. First let

$$\boldsymbol{U}^* = \begin{pmatrix} \boldsymbol{a}_{3\times 1} & \mathbf{0}_{3\times 1} \\ \mathbf{0}_{2\times 1} & \boldsymbol{b}_{2\times 1} \end{pmatrix} \in \mathbb{R}^{5\times 2}$$

where $\boldsymbol{a}$, $\boldsymbol{b}$ are vectors of independent standard normal random numbers. When $p = 10$, we obtain $\boldsymbol{U}_k \in \mathbb{R}^{5\times 2}$ by stacking two copies of independently generated $\boldsymbol{U}^*$'s, one on top of the other. For other dimensions $p = 20, 100$ similar procedure is followed to generate $\mathbf{U}_k$. Finally, we standardize these constructed matrices into orthonormal matrices (dividing each column by its length).

When $(p_1, p_2, p_3) = (10, 10, 10)$ and $(20, 20, 20)$, all $\boldsymbol{U}_k$'s are set to be sparse (and generated independently as described above). When $(p_1, p_2, p_3) = (100, 100, 3)$, $\boldsymbol{U}_1$ and $\boldsymbol{U}_2$ are sparse, while $\boldsymbol{U}_3$ is non-sparse (due to that its size is small) and generated as in Section 4.1. Accordingly, we use the penalty on $\|\mathbf{U}_1\|_1 \|\mathbf{U}_2\|_1$. The core tensors, the covariates and the random errors are generated in the same way as before. We again compare the performances with lasso. Since the number of free parameters is not straightforward to define for a sparse and orthogonal matrix, we select the ranks by minimizing the out of sample prediction error on independently generated data. Table 2 reports the estimation error and Table 3 reports the average selected ranks when $n = 400$. As in the nonsparse case, the proposed method outperforms lasso. In particular, lasso fails when $(p_1, p_2, p_3) = (20, 20, 20)$, with the estimation errors almost the same as the Frobenius norm of the true coefficients, but our approach still works. These results demonstrate that for a given finite sample with limited

			$\tau = 0.25$					
			$\sigma = 0.5$			$\sigma = 1$		
(p_1, p_2, p_3)	rank	n	LASSO	Proposed		LASSO	Proposed	
			EE	EE	rank	EE	EE	rank
(5,5,5)	(2,2,2)	1000	0.23(0.02)	0.11(0.02)	1.00	0.47(0.04)	0.22(0.04)	1.00
		2000	0.15(0.01)	0.07(0.01)	1.00	0.32(0.02)	0.16(0.02)	1.00
	(3,3,3)	1000	0.24(0.02)	0.14(0.02)	1.00	0.43(0.03)	0.29(0.03)	1.00
		2000	0.19(0.01)	0.10(0.01)	1.00	0.29(0.02)	0.20(0.02)	1.00
(10,10,10)	(2,2,2)	1000	1.70(0.13)	0.16(0.02)	0.97	3.38(0.27)	0.33(0.03)	0.96
		2000	0.55(0.02)	0.11(0.01)	1.00	1.18(0.05)	0.23(0.02)	1.00
	(3,3,3)	1000	1.93(0.16)	0.21(0.02)	1.00	3.36(0.23)	0.42(0.03)	1.00
		2000	0.52(0.02)	0.14(0.01)	1.00	1.11(0.04)	0.29(0.02)	1.00
			$\tau = 0.5$					
			$\sigma = 0.5$			$\sigma = 1$		
(p_1, p_2, p_3)	rank	n	LASSO	Proposed		LASSO	Proposed	
			EE	EE	rank	EE	EE	rank
(5,5,5)	(2,2,2)	1000	0.21(0.02)	0.10(0.02)	1.00	0.42(0.03)	0.20(0.03)	1.00
		2000	0.14(0.01)	0.07(0.01)	1.00	0.30(0.02)	0.14(0.02)	1.00
	(3,3,3)	1000	0.19(0.01)	0.13(0.01)	1.00	0.47(0.03)	0.27(0.03)	1.00
		2000	0.13(0.09)	0.09(0.01)	1.00	0.33(0.02)	0.19(0.02)	1.00
(10,10,10)	(2,2,2)	1000	0.94(0.15)	0.15(0.01)	1.00	1.45(0.18)	0.30(0.03)	1.00
		2000	0.53(0.02)	0.10(0.01)	1.00	1.11(0.04)	0.21(0.02)	1.00
	(3,3,3)	1000	1.88(0.15)	0.19(0.01)	1.00	3.37(0.23)	0.39(0.03)	1.00
		2000	0.51(0.02)	0.13(0.01)	1.00	1.06(0.03)	0.27(0.02)	1.00
			$\tau = 0.75$					
			$\sigma = 0.5$			$\sigma = 1$		
(p_1, p_2, p_3)	rank	n	LASSO	Proposed		LASSO	Proposed	
			EE	EE	rank	EE	EE	rank
(5,5,5)	(2,2,2)	1000	0.21(0.01)	0.11(0.01)	1.00	0.46(0.03)	0.22(0.04)	1.00
		2000	0.15(0.01)	0.08(0.01)	1.00	0.31(0.02)	0.16(0.02)	1.00
	(3,3,3)	1000	0.20(0.01)	0.14(0.02)	1.00	0.43(0.03)	0.29(0.03)	1.00
		2000	0.14(0.01)	0.10(0.01)	1.00	0.30(0.02)	0.21(0.02)	1.00
(10,10,10)	(2,2,2)	1000	1.70(0.13)	0.17(0.02)	0.99	3.40(0.28)	0.33(0.03)	0.99
		2000	0.55(0.02)	0.11(0.01)	1.00	1.18(0.04)	0.23(0.02)	1.00
	(3,3,3)	1000	1.94(0.17)	0.21(0.02)	1.00	3.40(0.24)	0.41(0.03)	1.00
		2000	0.52(0.02)	0.14(0.01)	1.00	1.11(0.04)	0.29(0.02)	1.00

Table 1: Estimation errors and the percentage of times selecting the true ranks for tensor quantile regression. Numbers in the parentheses denote the standard errors.

		$\tau = 0.25$			
		$\sigma = 0.5$		$\sigma = 1$	
(p_1, p_2, p_3)	n	LASSO	Proposed	LASSO	Proposed
(10,10,10)	200	2.40(0.72)	0.56(0.07)	2.52(0.67)	0.93(0.39)
	400	1.82(0.56)	0.36(0.04)	2.05(0.55)	0.51(0.06)
	800	1.44(0.57)	0.26(0.03)	1.70(0.48)	0.37(0.04)
(20,20,20)	200	2.97(0.63)	1.99(0.70)	3.01(0.62)	2.30(0.68)
	400	2.88(0.61)	0.62(0.11)	2.93(0.59)	0.94(0.20)
	800	2.66(0.54)	0.38(0.03)	2.73(0.54)	0.57(0.10)
(100,100,3)	200	3.02(0.73)	3.01(0.73)	3.04(0.73)	3.05(0.72)
	400	3.02(0.72)	2.16(0.64)	3.05(0.71)	2.38(0.63)
	800	2.98(0.71)	1.80(0.85)	3.01(0.69)	2.15(0.77)
		$\tau = 0.5$			
		$\sigma = 0.5$		$\sigma = 1$	
(p_1, p_2, p_3)	n	LASSO	Proposed	LASSO	Proposed
(10,10,10)	200	2.82(0.90)	0.38(0.07)	2.91(0.84)	0.84(0.37)
	400	2.10(0.71)	0.24(0.02)	2.32(0.70)	0.48(0.06)
	800	1.36(0.44)	0.17(0.02)	1.63(0.41)	0.35(0.04)
(20,20,20)	200	3.42(0.70)	1.61(0.70)	3.55(0.74)	2.08(0.66)
	400	3.20(0.86)	0.44(0.27)	3.25(0.66)	0.84(0.23)
	800	2.89(0.64)	0.25(0.02)	2.78(0.81)	0.52(0.07)
(100,100,3)	200	3.52(0.87)	3.03(0.70)	3.58(0.88)	3.10(0.74)
	400	3.32(0.87)	2.06(0.68)	3.45(0.86)	2.30(0.65)
	800	3.14(0.81)	1.56(0.89)	3.20(0.81)	2.01(0.76)
		$\tau = 0.75$			
		$\sigma = 0.5$		$\sigma = 1$	
(p_1, p_2, p_3)	n	LASSO	Proposed	LASSO	Proposed
(10,10,10)	200	2.41(0.69)	0.59(0.10)	2.54(0.66)	0.85(0.17)
	400	1.87(0.63)	0.36(0.04)	2.02(0.51)	0.54(0.07)
	800	1.54(0.63)	0.26(0.02)	1.70(0.53)	0.37(0.03)
(20,20,20)	200	2.97(0.63)	1.88(0.66)	3.00(0.62)	2.21(0.64)
	400	2.88(0.61)	0.65(0.22)	2.94(0.59)	0.93(0.20)
	800	2.66(0.55)	0.38(0.03)	2.73(0.54)	0.57(0.12)
(100,100,3)	200	3.02(0.73)	3.00(0.72)	3.04(0.71)	3.05(0.73)
	400	3.02(0.72)	2.17(0.67)	3.04(0.71)	2.40(0.66)
	800	2.98(0.70)	1.83(0.83)	3.02(0.70)	2.14(0.80)

Table 2: Estimation errors for sparse tensor quantile regression. Numbers in the parentheses denote the standard errors.

data, further dimension reduction mechanism brought about by tensor decomposition can further improve estimation efficiency compared to using sparsity alone.

		$\tau = 0.25$					
(p_1, p_2, p_3)	(r_1, r_2, r_3)	$\sigma = 0.5$			$\sigma = 1$		
		r_1	r_2	r_3	r_1	r_2	r_3
(10,10,10)	(2,2,2)	2.02(0.14)	2.02(0.14)	2.06(0.24)	2.00(0.29)	2.00(0.29)	2.04(0.35)
(20,20,20)	(2,2,2)	1.98(0.38)	2.00(0.35)	2.12(0.39)	1.78(0.42)	1.82(0.39)	1.98(0.51)
(100,100,3)	(2,2,2)	1.40(0.61)	1.56(0.61)	2.62(0.60)	1.44(0.64)	1.68(0.62)	2.62(0.53)
		$\tau = 0.5$					
(p_1, p_2, p_3)	(r_1, r_2, r_3)	$\sigma = 0.5$			$\sigma = 1$		
		r_1	r_2	r_3	r_1	r_2	r_3
(10,10,10)	(2,2,2)	2.02(0.14)	2.02(0.14)	2.08(0.27)	2.02(0.25)	2.02(0.25)	2.10(0.36)
(20,20,20)	(2,2,2)	1.94(0.24)	1.94(0.24)	2.04(0.28)	1.80(0.40)	1.84(0.37)	2.02(0.47)
(100,100,3)	(2,2,2)	1.42(0.70)	1.60(0.70)	2.70(0.54)	1.60(0.83)	1.84(0.77)	2.70(0.46)
		$\tau = 0.75$					
(p_1, p_2, p_3)	(r_1, r_2, r_3)	$\sigma = 0.5$			$\sigma = 1$		
		r_1	r_2	r_3	r_1	r_2	r_3
(10,10,10)	(2,2,2)	1.98(0.14)	1.98(0.14)	2.12(0.33)	1.94(0.31)	1.94(0.31)	2.02(0.32)
(20,20,20)	(2,2,2)	1.94(0.31)	1.94(0.31)	2.00(0.40)	1.82(0.48)	1.82(0.48)	1.92(0.49)
(100,100,3)	(2,2,2)	1.30(0.61)	1.54(0.65)	2.72(0.54)	1.60(0.78)	1.82(0.72)	2.76(0.52)

Table 3: Mean and standard error of the estimated ranks for sparse Tucker tensor quantile regression when $n = 400$. The true rank is $(2, 2, 2)$.

Then we consider the situation with unequal ranks for different modes of the tensor, which is set to be $(2, 3, 4)$. The factor matrices $\boldsymbol{U}_1 \in \mathbb{R}^{p_1 \times 2}$, $\boldsymbol{U}_2 \in \mathbb{R}^{p_2 \times 3}$ and $\boldsymbol{U}_3 \in \mathbb{R}^{p_3 \times 4}$ are respectively generated by stacking $\boldsymbol{U}^*$ (as defined before) and

$$\boldsymbol{U}_2^* = \begin{pmatrix} \boldsymbol{a}_{3\times 2} & \boldsymbol{0}_{3\times 1} \\ \boldsymbol{0}_{2\times 2} & \boldsymbol{b}_{2\times 1} \end{pmatrix} \in \mathbb{R}^{5\times 3}, \boldsymbol{U}_3^* = \begin{pmatrix} \boldsymbol{a}_{3\times 2} & \boldsymbol{0}_{3\times 2} \\ \boldsymbol{0}_{2\times 2} & \boldsymbol{b}_{2\times 2} \end{pmatrix} \in \mathbb{R}^{5\times 4}.$$

The orthogonal matrices $\boldsymbol{a}_{3\times 2}$ and $\boldsymbol{b}_{2\times 2}$ are obtained by QR decomposition as before. Figure 1 and Figure 2 compare the proposed method and lasso quantile regression under different choice of sample sizes and dimensions. As expected, the estimation errors increase as dimensions increases, and as the sample size n decreases, and the errors are smaller than lasso quantile regression.

Moreover, as suggested by a reviewer, we consider the alternative approach of simply imposing convex regularizations on the coefficient tensor $\boldsymbol{\mathcal{A}}$ to encourage low-dimensional structure. To be specific, we impose the nuclear norm regularization on the matricized coefficient tensors to encourage low-rankness. The objective function is

$$\frac{1}{n}\sum_{i=1}^{n} \rho_\tau \left(y_i - \langle \boldsymbol{\mathcal{A}}, \boldsymbol{\mathcal{X}}_i \rangle\right) + \frac{\lambda_1}{K}\sum_{k=1}^{K} \|\boldsymbol{\mathcal{A}}_{(k)}\|_*, \tag{9}$$

where λ_1 is the tuning parameter, and $\|\cdot\|_*$ stands for the matrix nuclear norm. We can also add an entry-wise ℓ_1 penalty on $\boldsymbol{\mathcal{A}}$ to encourage sparsity, resulting in

$$\frac{1}{n}\sum_{i=1}^{n} \rho_\tau \left(y_i - \langle \boldsymbol{\mathcal{A}}, \boldsymbol{\mathcal{X}}_i \rangle\right) + \frac{\lambda_1}{K}\sum_{k=1}^{K} \|\boldsymbol{\mathcal{A}}_{(k)}\|_* + \lambda_2 \sum_{j_1=1}^{p_1}\sum_{j_2=1}^{p_2}\cdots\sum_{j_K=1}^{p_K} |\boldsymbol{\mathcal{A}}_{j_1 j_2 \ldots j_K}|. \tag{10}$$

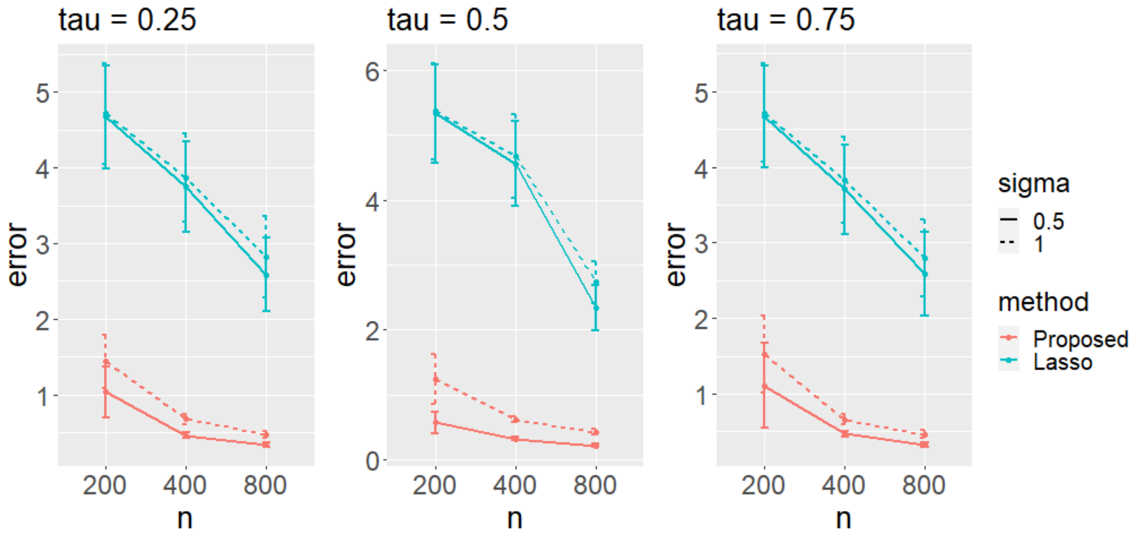


Figure 1: Estimation errors based on 50 replications under quantile level $\tau = 0.25$, 0.5 and 0.75, when true ranks are $(2, 3, 4)$ and dimension $p_1 = p_2 = p_3 = 10$. The error bars represent ± one standard deviation.

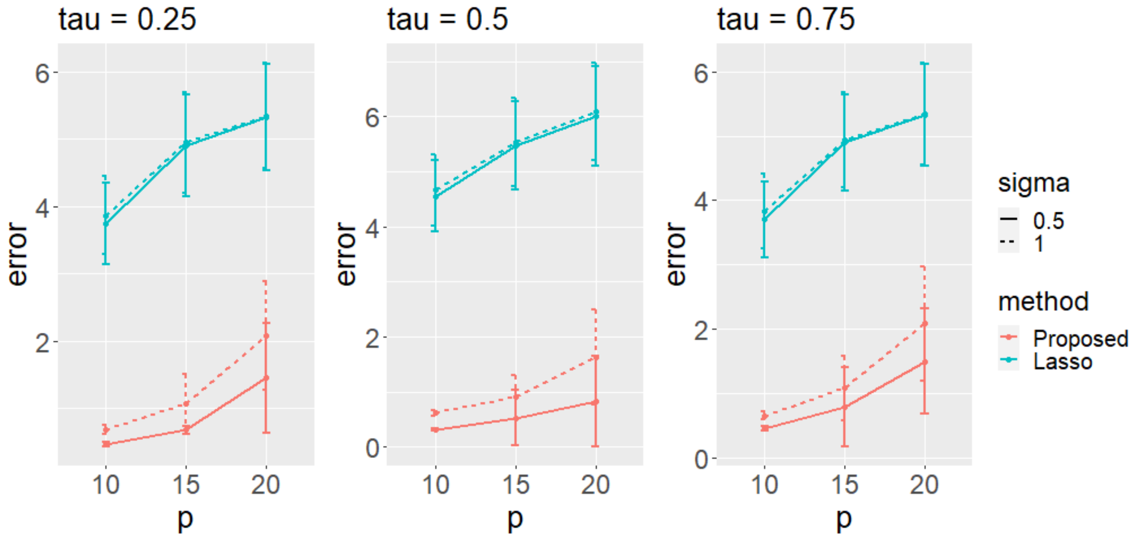


Figure 2: Estimation error based on 50 replications under quantile levels $\tau = 0.25$, 0.5 and 0.75, when the true rank is $(2, 3, 4)$ and sample size $n = 400$. The dimension of the coefficient tensor is (p, p, p). The error bars represent ± one standard deviation.

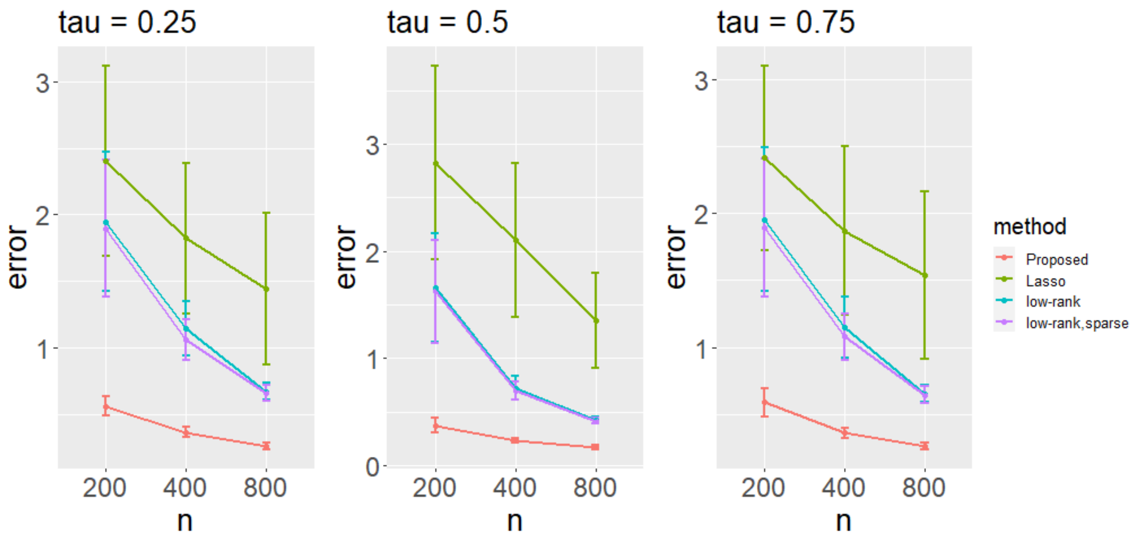


Figure 3: Mean of estimation error based on 50 replications under quantile level $\tau = 0.25$, 0.5 and 0.75, when the dimension $p_1 = p_2 = p_3 = 10$, ranks $(r_1, r_2, r_3) = (2, 2, 2)$ and noise level $\sigma = 0.5$. The error bars represent ± one standard deviation.

Note that the sparsity regularization in (10) is not on the factor matrices $\mathbf{U}_k$ and thus represents a different model of sparsity. The minimizer of (9) and (10) are computed by the proximal gradient method. For this experiment, we still generated datasets as before and take ranks $(r_1, r_2, r_3) = (2, 2, 2)$, dimensions $(p_1, p_2, p_3) = (10, 10, 10)$, sample size $n = 200, 400, 800$ and noise level $\sigma = 0.5$. Figure 3 shows the error of the estimate computed by (9), (10), lasso and the proposed method. Although in this experiment, our method performs better than convex regularization, which is certainly as expected, we are not saying the convex penalization approach is always worse. Convex approach as an alternative approach also deserves careful theoretical and computational study in the convex of quantile tensor regression, but a detailed investigation is outside the scope of the current paper.

5. Application to PETS 2009 dataset

In this section, we show the effectiveness of the proposed method by performing experiments on PETS 2009 dataset (`http://www.cvg.reading.ac.uk/PETS2009`). The dataset contains crowd images recorded at Whiteknights Campus, University of Reading, UK. Our goal is to estimate the number of people by fitting a model that takes the color images as tensor covariates ($136 \times 186 \times 3$ tensor). Several views of the same scene are available and we only use View 1 in this analysis. Figure 4 shows two example images of View 1 in the dataset. The ground-truth count is obtained from Chan et al. (2009).

We fit a quantile regression model using 662 images from timestamps 13-57, 13-59 and the first 200 images with timestamp 14-03. The remaining 213 images of 14-03 are used as the test set. The raw count is taken as the response. We also tried some transformations on

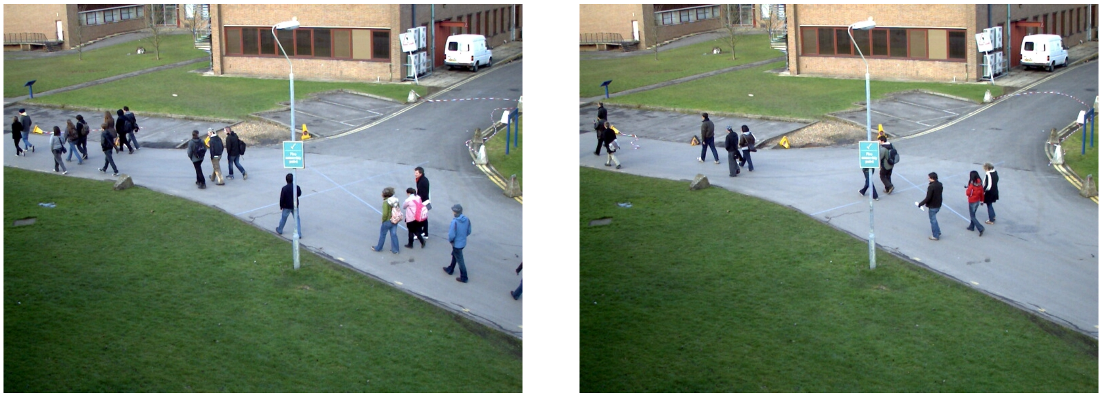
	$\tau = 0.25$			
	Training error		Prediction error	
	Lasso	Proposed	Lasso	Proposed
Raw	0.18	0.09	2.49	2.18
Square root	0.47	0.07	1.26	0.89
	$\tau = 0.5$			
	Training error		Prediction error	
	Lasso	Proposed	Lasso	Proposed
Raw	0.21	0.11	1.96	1.09
Square root	0.35	0.19	1.25	0.78
	$\tau = 0.75$			
	Training error		Prediction error	
	Lasso	Proposed	Lasso	Proposed
Raw	0.39	0.10	1.04	0.91
Square root	0.58	0.15	0.98	0.72

Table 4: Training and prediction error for lasso and the proposed method on the test set, either directly using the count as the response, or using square-root transformation.

the response from the Box-Cox family and found that the square-root transformation works well. We examine the performances of the proposed sparse tensor regression approach on predicting the 0.25, 0.5 and 0.75 quantile of the counts on the test set. Ranks are chosen from all combinations of ranks up to 4 (except r_3 is of course bounded by 3). Since the dimension of mode-3 is small, we only impose penalty on factor matrices $\boldsymbol{U}_1$ and $\boldsymbol{U}_2$. The tuning parameter and ranks were set by 10-fold cross-validation, resulting in low-rank estimators with rank $(3, 3, 3)$. Using the low-rank structure, the number of parameters is reduced by about 98.7%. Table 4 shows the training errors and prediction errors in terms of quantile loss (for prediction, responses are transformed back to the original scale after model fitting on the transformed response). As in simulations, we compared the proposed method with lasso quantile regression which vectorizes the image. It can be seen that the proposed method leads to smaller errors. Figure 4 shows two examples with the predicted quantile values calculated by the proposed method (with square-root transformation).

6. Conclusion

In this paper, we propose a Tucker decomposition-based estimation approach for quantile regression with tensor covariates. The motivation of our work is the increasing demand for analysing tensor valued data such as images, and the lack of studies on quantile regression for this data type. The high dimensionality which often emerges in tensor regression is reduced by imposing a low-rank approximation and then applying Tucker decomposition. In addition, when the dimension is very large, we introduce a sparse Tucker decomposition to further reduce the number of parameters. We establish the asymptotic distribution for the nonsparse scenario and the bound on estimation error for the sparse scenario.



(a) The true count in this image is 17. The predicted 0.25, 0.5 and 0.75 quantiles are 14.97, 17.15 and 19.90, respectively.

(b) The true count in this image is 11. The predicted 0.25, 0.5 and 0.75 quantiles are 9.07, 10.70 and 12.88, respectively.

Figure 4: Examples in the test set with predicted quantiles calculated by the proposed method.

In the quantile regression setting, it is typical to consider a scalar response. However, one can also consider the case where the response is a tensor. In this case, the intercept is a tensor of the same size as the response. Our results can be easily extended to this case in a straightforward way to predict each response's quantile.

For both nonsparse and sparse Tucker quantile regression, we develop alternating algorithms to estimate the coefficient tensor. The convergence result for the nonconvex problem is generally an open problem with success only in some special problems. Developing efficient and provably convergent algorithms for the model is a challenging issue which can be investigated in the future. Our implementation of the algorithm based on R can be obtained from `https://github.com/WenqiLu/QuantileTensorReg`. We note that for probably more efficient implementation in Python for example, the library TensorLy (Kossaifi et al., 2019) provides a high-level API, and allows the model to be run on multiple CPU and GPU machines.

Acknowledgments

The authors sincerely thank the editor Professor Animashree Anandkumar, and four anonymous reviewers for their insightful comments that improved the manuscript. The research of Lian is supported by Hong Kong RGC general research fund 11301718 and 11300519, and by Project 11871411 from NSFC and the Shenzhen Research Institute, City University of Hong Kong. The research of Zhu is supported by National Natural Science Foundation of China (112071087, 11731011).

Appendix A.

In the proofs C denotes a generic constant whose value may change even on the same line.

A.1 Proof of Theorem 1

In the main text, the true parameters are simply denoted by $\boldsymbol{\mathcal{A}}, \mathbf{U}, \mathbf{g}$, etc. For this proof only, we will use subscript 0 to denote the true parameters which will make the presentation more clear. Using our notations,

$$\widehat{\boldsymbol{\mathcal{A}}} = \arg\min \sum_{i=1}^{n} \rho_\tau\left(y_i - \langle \boldsymbol{\mathcal{G}} \times_1 \boldsymbol{U}_1 \times_2 \boldsymbol{U}_2 \cdots \times_K \boldsymbol{U}_K, \boldsymbol{\mathcal{X}}_i\rangle\right)$$

can be rewritten as

$$\widehat{\mathbf{h}} = \mathbf{h}(\widehat{\boldsymbol{\phi}}) = \underset{\mathbf{h}=\mathbf{h}(\boldsymbol{\phi})}{\arg\min} \sum_{i=1}^{n} \rho_\tau\big(y_i - \boldsymbol{x}_i^T \boldsymbol{h}\big). \tag{11}$$

We first establish consistency. Let $\boldsymbol{h}_0 = \boldsymbol{h}(\boldsymbol{\phi}_0)$ be the true value of $\boldsymbol{h}$. Define

$$\begin{aligned} Q_n(\boldsymbol{h}) &= \frac{1}{n}\sum_{i=1}^{n} \big[\rho_\tau(y_i - \boldsymbol{x}_i^T \boldsymbol{h}) - \rho_\tau(y_i - \boldsymbol{x}_i^T \boldsymbol{h}_0)\big], \\ Q(\boldsymbol{h}) &= E\big\{\rho_\tau(y - \boldsymbol{x}^T \boldsymbol{h}) - \rho_\tau(y - \boldsymbol{x}^T \boldsymbol{h}_0)\big\}. \end{aligned}$$

Since $|\rho_\tau(y-\boldsymbol{x}^T\boldsymbol{h}) - \rho_\tau(y-\boldsymbol{x}^T\boldsymbol{h}')| \leq \max(\tau, 1-\tau)\|\boldsymbol{x}\|\|\boldsymbol{h}-\boldsymbol{h}'\|$ for all $\boldsymbol{h}, \boldsymbol{h}'$, by Example 19.7 in van der Vaart (1998), the class $\big\{\rho_\tau(y-\boldsymbol{x}^T\boldsymbol{h}) - \rho_\tau(y-\boldsymbol{x}^T\boldsymbol{h}_0) : \boldsymbol{h} \in \mathcal{H}\big\}$ is Glibenko-Cantelli. Then the first condition of Theorem 5.7 in van der Vaart (1998), that is $\sup_{\boldsymbol{h}\in\mathcal{H}} |Q_n(\mathbf{h}) - Q(\mathbf{h})| = o_p(1)$, is verified, where $\mathcal{H}$ denotes the parameter space for $\mathbf{h}$ which is assumed to be bounded by condition C3.

Using Knight's identity, we have

$$\begin{aligned} Q(\boldsymbol{h}) - Q(\boldsymbol{h}_0) &= E\int_0^{\boldsymbol{x}^T(\boldsymbol{h}-\boldsymbol{h}_0)} (F_{\varepsilon|\boldsymbol{x}}(t) - F_{\varepsilon|\boldsymbol{x}}(0))\,\mathrm{d}t \\ &\geq \frac{1}{2}\underline{f}\delta_{\min}(\boldsymbol{D}_0)\|\boldsymbol{h}-\boldsymbol{h}_0\|^2 - \frac{1}{6}\overline{f'}E[\|\boldsymbol{x}\|^3]\|\boldsymbol{h}-\boldsymbol{h}_0\|^3, \end{aligned}$$

where $\underline{f}$ denotes the lower bound for $f_{\varepsilon|\mathbf{x}}(0)$ and $\overline{f'}$ denotes the upper bound for $f'_{\varepsilon|\mathbf{x}}(.)$.

Let $c = \frac{3\delta_{\min}(\boldsymbol{D}_0)\underline{f}}{2\overline{f'}E\|\boldsymbol{x}\|^3}$, it follows that when $\|\boldsymbol{h}-\boldsymbol{h}_0\| \leq c$,

$$Q(\boldsymbol{h}) - Q(\boldsymbol{h}_0) \geq \frac{1}{4}\underline{f}\delta_{\min}(\boldsymbol{D}_0)\|\boldsymbol{h}-\boldsymbol{h}_0\|^2. \tag{12}$$

When $\|\boldsymbol{h}-\boldsymbol{h}_0\| > c$, let $\boldsymbol{h}' = c'\mathbf{h} + (1-c')\boldsymbol{h}_0$ with $c' = \frac{c}{\|\boldsymbol{h}-\boldsymbol{h}_0\|}$, then $\|\boldsymbol{h}'-\boldsymbol{h}_0\| = c$, and by the convexity of Q, we have

$$\begin{aligned} Q(\boldsymbol{h}) - Q(\boldsymbol{h}_0) &\geq \frac{\|\boldsymbol{h}-\boldsymbol{h}_0\|}{c}\big(Q(\boldsymbol{h}') - Q(\boldsymbol{h}_0)\big) \\ &\geq \frac{c}{4}\underline{f}\delta_{\min}(\boldsymbol{D}_0)\|\boldsymbol{h}-\boldsymbol{h}_0\|. \end{aligned}$$

Therefore, the second condition in Theorem 5.7 in van der Vaart (1998) $\inf_{\|\boldsymbol{h}-\boldsymbol{h}_0\|\geq\varepsilon}\big\{Q(\boldsymbol{h})-Q(\boldsymbol{h}_0)\big\}>0$ is verified. By Theorem 5.7 in van der Vaart (1998), we have the sequence $h(\widehat{\boldsymbol{\phi}})$ converges in probability to $\boldsymbol{h}_0$.

Next we establish the convergence rate. The first condition of Theorem 5.52 in van der Vaart (1998) is verified in (12). For every sufficiently small $\delta>0$, let $\mathcal{F}=\{\rho_\tau(y-\boldsymbol{x}^T\boldsymbol{h})-\rho_\tau(y-\boldsymbol{x}^T\boldsymbol{h}_0);\|\boldsymbol{h}-\boldsymbol{h}_0\|<\delta\}$. This class has an envelop function δm where $m=\max(\tau,1-\tau)\|\boldsymbol{x}\|$. By Corollary 19.35 and Example 19.7 in van der Vaart (1998), we get

$$\begin{aligned}E\sup_{f\in\mathcal{F}}|\sqrt{n}(P_n(f)-Pf)| &\lesssim J_{[\,]}\big(\|m\delta\|_{p,2},\mathcal{F},L_2(P)\big)\\ &\lesssim\int_0^{\|m\|_{p,2}\delta}\|m\|_{p,2}\sqrt{\log(\frac{\delta}{\varepsilon})^p}\,\mathrm{d}\varepsilon,\end{aligned}$$

where $\|m\|_{p,2}=(E|m|^2)^{1/2}$, and $p=\prod_{k=1}^K p_k$. Change the variables in the integral to see that this is a multiple of δ. Then by Theorem 5.52 in van der Vaart (1998), we have $\|\mathbf{h}(\widehat{\boldsymbol{\phi}})-\boldsymbol{h}_0\|=O_p(n^{-1/2})$.

Furthermore, we can prove the following uniform convergence

$$\begin{aligned}\sup_{\|\boldsymbol{h}-\boldsymbol{h}_0\|\leq C/\sqrt{n}}|\sum_{i=1}^n\rho_\tau(y_i-\boldsymbol{x}_i^T\boldsymbol{h})-\sum_{i=1}^n\rho_\tau(y_i-\boldsymbol{x}_i^T\widehat{\boldsymbol{h}}_{QR})&\\ -\frac{n}{2}(\boldsymbol{h}-\widehat{\boldsymbol{h}}_{QR})^T\boldsymbol{D}_1(\boldsymbol{h}-\widehat{\boldsymbol{h}}_{QR})|=o_p(1),&\end{aligned}\tag{13}$$

where $\widehat{\boldsymbol{h}}_{QR}$ is the vectorized quantile regression estimate $\mathrm{vec}(\widehat{\boldsymbol{\mathcal{A}}}_{QR})$ with $\widehat{\boldsymbol{\mathcal{A}}}_{QR}=\arg\min\sum_{i=1}^n\rho_\tau\big(y_i-\boldsymbol{x}_i^T\mathrm{vec}(\boldsymbol{\mathcal{A}})\big)$ (without the rank constraint).

In fact, by Lemma 19.31 of van der Vaart (1998), we have

$$\begin{aligned}\sup_{\|\boldsymbol{h}-\boldsymbol{h}_0\|=C/\sqrt{n}}\Bigg|\sum_{i=1}^n\rho_\tau(y_i-\boldsymbol{x}_i^T\boldsymbol{h})-\sum_{i=1}^n\rho_\tau(y_i-\boldsymbol{x}_i^T\boldsymbol{h}_0)&\\ -(\boldsymbol{h}-\boldsymbol{h}_0)^T\sum_{i=1}^n\boldsymbol{x}_i\big(\tau-I(y_i-\boldsymbol{x}_i^T\boldsymbol{h}_0)\big)&\\ -nE\big[\rho_\tau(Y-\boldsymbol{x}^T\boldsymbol{h})-\rho_\tau(Y-\boldsymbol{x}^T\boldsymbol{h}_0)\big]\Bigg|=o_p(1),&\end{aligned}\tag{14}$$

and

$$\begin{aligned}\Bigg|\sum_{i=1}^n\rho_\tau(y_i-\boldsymbol{x}_i^T\widehat{\boldsymbol{h}}_{QR})-\sum_{i=1}^n\rho_\tau(y_i-\boldsymbol{x}_i^T\boldsymbol{h}_0)-(\widehat{\boldsymbol{h}}_{QR}-\boldsymbol{h}_0)^T\sum_{i=1}^n\boldsymbol{x}_i\big(\tau-I(y_i-\boldsymbol{x}_i^T\boldsymbol{h}_0)\big)&\\ -nE\big[\rho_\tau(Y-\boldsymbol{x}^T\widehat{\boldsymbol{h}}_{QR})-\rho_\tau(Y-\boldsymbol{x}^T\boldsymbol{h}_0)\big]\Bigg|=o_p(1).&\end{aligned}\tag{15}$$

By Knight's identity, we have

$$\begin{aligned}E\big[\rho_\tau(Y-\boldsymbol{x}^T\boldsymbol{h})-\rho_\tau(Y-\boldsymbol{x}^T\boldsymbol{h}_0)\big]&=E\int_0^{\boldsymbol{x}^T(\boldsymbol{h}-\boldsymbol{h}_0)}(F_{\varepsilon|\boldsymbol{x}}(t)-F_{\varepsilon|\boldsymbol{x}}(0))\,\mathrm{d}t\\ &=\frac{1}{2}(\boldsymbol{h}-\boldsymbol{h}_0)^T\boldsymbol{D}_1(\boldsymbol{h}-\boldsymbol{h}_0)+o_p(\|\boldsymbol{h}-\boldsymbol{h}_0\|^2).\end{aligned}\tag{16}$$

It follows that

$$\sup_{\|\boldsymbol{h}-\boldsymbol{h}_0\|=C/\sqrt{n}} \left| nE\left[\rho_\tau(Y-\boldsymbol{x}^T\boldsymbol{h})-\rho_\tau(Y-\boldsymbol{x}^T\boldsymbol{h}_0)\right]-\frac{n}{2}(\boldsymbol{h}-\boldsymbol{h}_0)^T\boldsymbol{D}_1(\boldsymbol{h}-\boldsymbol{h}_0)\right| = o_p(1), \quad (17)$$

and

$$\left| nE\left[\rho_\tau(Y-\boldsymbol{x}^T\widehat{\boldsymbol{h}}_{QR})-\rho_\tau(Y-\boldsymbol{x}^T\boldsymbol{h}_0)\right]-\frac{n}{2}(\widehat{\boldsymbol{h}}_{QR}-\boldsymbol{h}_0)^T\boldsymbol{D}_1(\widehat{\boldsymbol{h}}_{QR}-\boldsymbol{h}_0)\right| = o_p(1), \quad (18)$$

Combining (14) - (18), we have

$$\begin{aligned}\sup_{\|\boldsymbol{h}-\boldsymbol{h}_0\|=C/\sqrt{n}} \Bigg| \sum_{i=1}^n \rho_\tau(y_i-\boldsymbol{x}_i^T\boldsymbol{h})-\sum_{i=1}^n \rho_\tau(y_i-\boldsymbol{x}_i^T\widehat{\boldsymbol{h}}_{QR}) \\ -(\boldsymbol{h}-\widehat{\boldsymbol{h}}_{QR})^T\sum_{i=1}^n \boldsymbol{x}_i\big(\tau-I(y_i-\boldsymbol{x}_i^T\boldsymbol{h}_0)\big) \qquad (19) \\ -\frac{n}{2}(\boldsymbol{h}-\boldsymbol{h}_0)^T\boldsymbol{D}_1(\boldsymbol{h}-\boldsymbol{h}_0)+\frac{n}{2}(\widehat{\boldsymbol{h}}_{QR}-\boldsymbol{h}_0)^T\boldsymbol{D}_1(\widehat{\boldsymbol{h}}_{QR}-\boldsymbol{h}_0)\Bigg| = o_p(1).\end{aligned}$$

Then equation (13) can be obtained from (19) by applying the Bahadur representation for the classical linear quantile estimator $\sqrt{n}(\widehat{\boldsymbol{h}}_{QR}-\boldsymbol{h}_0)=-\boldsymbol{D}_1^{-1}\frac{1}{\sqrt{n}}\sum_{i=1}^n \boldsymbol{x}_i\big(\tau-I(y_i-\boldsymbol{x}_i^T\boldsymbol{h}_0<0)\big)+o_p(1)$.

Define

$$F(\widehat{\boldsymbol{h}}_{QR},\boldsymbol{h})=\frac{n}{2}(\boldsymbol{h}-\widehat{\boldsymbol{h}}_{QR})^T\boldsymbol{D}_1(\boldsymbol{h}-\widehat{\boldsymbol{h}}_{QR}),$$

and denote by $\tilde{\boldsymbol{h}}=\mathbf{h}(\tilde{\boldsymbol{\phi}})$ as the minimizer of $F(\widehat{\boldsymbol{h}}_{QR},\boldsymbol{h})$, then by Proposition 4.1 in Shapiro (1986), it has asymptotic normality

$$\sqrt{n}(\tilde{\boldsymbol{h}}-\boldsymbol{h}_0)\xrightarrow{D} N(\mathbf{0},\boldsymbol{P}\boldsymbol{\Gamma}\boldsymbol{P}^T).$$

To complete the proof, we note (13) implies $\tilde{\boldsymbol{h}}$ and $\widehat{\boldsymbol{h}}$ are asymptotically equivalent. ■

Remark 2 *It is trivial to incorporate the intercept for this fixed-dimensional result. All derivations up to equation* (19) *has nothing to do with the tensor structure. We only need to add* μ *to* $\boldsymbol{h}$ *and add* 1 *to* $\boldsymbol{x}_i$*, and we still define* $\boldsymbol{D}_0=E(\boldsymbol{x}_i\boldsymbol{x}_i^T)$*,* $\boldsymbol{D}_1=E(\boldsymbol{x}_i\boldsymbol{x}_i^Tf_{\varepsilon|\boldsymbol{x}}(0))$ *and the asymptotic normality result remains the same.*

A.2 Proof of Theorem 2

Define $\widehat{\mathbf{U}}=(\widehat{\mathbf{U}}_K\widehat{\mathbf{O}}_K)\otimes\cdots\otimes(\widehat{\mathbf{U}}_1\widehat{\mathbf{O}}_1)$ and $\widehat{\mathbf{g}}=\mathrm{vec}(\widehat{\boldsymbol{\mathcal{G}}}\times_1\widehat{\mathbf{O}}_1^T\cdots\times_K\widehat{\mathbf{O}}_K^T)$, where the matrices $\widehat{\mathbf{O}}_k$ are defined in the statement of Lemma 2. Write $\mathrm{vec}(\widehat{\boldsymbol{\mathcal{A}}})=\widehat{\mathbf{U}}\widehat{\mathbf{g}}$. Let $\widehat{\boldsymbol{\Delta}}=\widehat{\boldsymbol{U}}\widehat{\boldsymbol{g}}-\boldsymbol{U}\boldsymbol{g}$, $\widehat{\boldsymbol{\Delta}}_{\boldsymbol{U}}=\widehat{\boldsymbol{U}}-\boldsymbol{U}$ and $\widehat{\boldsymbol{\Delta}}_{\boldsymbol{g}}=\widehat{\boldsymbol{g}}-\boldsymbol{g}$. Then $\widehat{\boldsymbol{\Delta}}=\widehat{\boldsymbol{\Delta}}_{\boldsymbol{U}}\widehat{\boldsymbol{g}}+\boldsymbol{U}\widehat{\boldsymbol{\Delta}}_{\boldsymbol{g}}$ and $\|\widehat{\boldsymbol{\mathcal{A}}}-\boldsymbol{\mathcal{A}}\|_F=\|\widehat{\boldsymbol{\Delta}}\|$. By the optimality of $\widehat{\boldsymbol{U}}$ and $\widehat{\boldsymbol{g}}$, we have

$$\frac{1}{n}\sum_{i=1}^n \rho_\tau\big(y_i-\boldsymbol{x}_i^T\widehat{\boldsymbol{U}}\widehat{\boldsymbol{g}}\big)+\lambda\|\widehat{\boldsymbol{U}}\|_1\le\frac{1}{n}\sum_{i=1}^n \rho_\tau\big(y_i-\boldsymbol{x}_i^T\boldsymbol{U}\boldsymbol{g}\big)+\lambda\|\boldsymbol{U}\|_1. \quad (20)$$

We first show that $\widehat{\mathbf{\Delta}} \in \mathbf{\Omega} := \{\mathbf{\Delta} : \|(\mathbf{\Delta}_{\mathbf{U}})_{S^c}\|_1 \le 3\|(\mathbf{\Delta}_{\mathbf{U}})_S\|_1 + \|\mathbf{\Delta}_{\mathbf{g}}\|_1\}$, where S is the support of $\mathbf{U}$ with size bounded by $\sqrt{s_1 s_2 \cdots s_K} =: s$. Let $e_i = \tau - I(y_i - \boldsymbol{x}_i^T \mathbf{U}\mathbf{g} \le 0)$. By convexity of ρ_τ, we have $Q_n(\widehat{\mathbf{U}}\widehat{\mathbf{g}}) := \frac{1}{n}(\sum_i \rho_\tau(y_i - \mathbf{x}_i\widehat{\mathbf{U}}\widehat{\mathbf{g}}) - \sum_i \rho_\tau(y_i - \mathbf{x}_i\mathbf{U}\mathbf{g})) \ge -\frac{1}{n}(\sum_{i=1}^n \boldsymbol{x}_i e_i)^T\widehat{\mathbf{\Delta}}$. Combining this with (20) we get,

$$-\frac{1}{n}(\sum_i \mathbf{x}_i e_i)(\widehat{\mathbf{\Delta}}_{\mathbf{U}}\widehat{\mathbf{g}}) - \frac{1}{n}(\sum_i \mathbf{x}_i e_i)\mathbf{U}\widehat{\mathbf{\Delta}}_{\mathbf{g}} \le Q_n(\widehat{\mathbf{U}}\widehat{\mathbf{g}}) \le \lambda\|\mathbf{U}\|_1 - \lambda\|\widehat{\mathbf{U}}\|_1.$$

By Theorem 1 of Belloni and Chernozhukov (2011), we have $\lambda/2 \ge \bar{g}\|\frac{1}{n}\sum_i \mathbf{x}_i e_i\|_\infty$ and $\lambda/2 \ge \|\frac{1}{n}\mathbf{U}^T\mathbf{x}_i e_i\|_\infty$. Using $\left|\frac{1}{n}(\sum_i \mathbf{x}_i e_i)(\widehat{\mathbf{\Delta}}_{\mathbf{U}}\widehat{\mathbf{g}})\right| \le \|\frac{1}{n}\sum_i \mathbf{x}_i e_i\|_\infty \|\widehat{\mathbf{\Delta}}_{\mathbf{U}}\widehat{\mathbf{g}}\|_1 \le \bar{g}\|\frac{1}{n}\sum_i \mathbf{x}_i e_i\|_\infty\|\widehat{\mathbf{\Delta}}_{\mathbf{U}}\|_1 \le (\lambda/2)\|\widehat{\mathbf{\Delta}}_{\mathbf{U}}\|_1$ and $\left|\frac{1}{n}(\sum_i \mathbf{x}_i e_i)\mathbf{U}\widehat{\mathbf{\Delta}}_{\mathbf{g}}\right| \le \|\frac{1}{n}\mathbf{U}^T\mathbf{x}_i e_i\|_\infty\|\widehat{\mathbf{\Delta}}_{\mathbf{g}}\|_1 \le (\lambda/2)\|\widehat{\mathbf{\Delta}}_{\mathbf{g}}\|_1$, we get

$$-(\lambda/2)\|\widehat{\mathbf{\Delta}}_{\mathbf{U}}\|_1 - (\lambda/2)\|\widehat{\mathbf{\Delta}}_{\mathbf{g}}\|_1 \le \lambda\|\mathbf{U}\|_1 - \lambda\|\widehat{\mathbf{U}}\|_1. \tag{21}$$

Using $\|\mathbf{U}\|_1 = \|\mathbf{U}_S\|_1$ and $\|\widehat{\mathbf{U}}\|_1 = \|\widehat{\mathbf{\Delta}}_{\mathbf{U}} + \mathbf{U}\|_1 = \|(\widehat{\mathbf{\Delta}}_{\mathbf{U}} + \mathbf{U})_S\|_1 + \|(\widehat{\mathbf{\Delta}}_{\mathbf{U}})_{S^c}\|_1 \ge \|\mathbf{U}_S\|_1 - \|(\widehat{\mathbf{\Delta}}_{\mathbf{U}})_S\|_1 + \|(\widehat{\mathbf{\Delta}}_{\mathbf{U}})_{S^c}\|_1$, the above implies

$$-(\lambda/2)\|\widehat{\mathbf{\Delta}}_{\mathbf{U}}\|_1 - (\lambda/2)\|\widehat{\mathbf{\Delta}}_{\mathbf{g}}\|_1 \le \lambda\|(\widehat{\mathbf{\Delta}}_{\mathbf{U}})_S\|_1 - \lambda\|(\widehat{\mathbf{\Delta}}_{\mathbf{U}})_{S^c}\|_1. \tag{22}$$

Using $\|\widehat{\mathbf{\Delta}}_{\mathbf{U}}\|_1 = \|(\widehat{\mathbf{\Delta}}_{\mathbf{U}})_S\|_1 + \|(\widehat{\mathbf{\Delta}}_{\mathbf{U}})_{S^c}\|_1$, the above is seen to be equivalent to $\widehat{\mathbf{\Delta}} \in \mathbf{\Omega}$.

Furthermore, due to $\widehat{\mathbf{\Delta}} \in \mathbf{\Omega}$, we have

$$\begin{aligned}
\|\widehat{\mathbf{\Delta}}\|_1 &\le \bar{g}\|\widehat{\mathbf{\Delta}}_{\mathbf{U}}\|_1 + \|\mathbf{U}\widehat{\mathbf{\Delta}}_{\mathbf{g}}\|_1 \\
&\le C\bar{g}\|(\widehat{\mathbf{\Delta}}_{\mathbf{U}})_S\|_1 + C\bar{g}\|\widehat{\mathbf{\Delta}}_{\mathbf{g}}\|_1 + \sqrt{r}\|\widehat{\mathbf{\Delta}}_{\mathbf{g}}\| \\
&\le C\bar{g}\sqrt{s}\|\widehat{\mathbf{\Delta}}_{\mathbf{U}}\|_F + C(\bar{g}+1)\sqrt{r}\|\widehat{\mathbf{\Delta}}_{\mathbf{g}}\| \\
&\le b_n\|\widehat{\mathbf{\Delta}}\| + d_n\|\widehat{\mathbf{\Delta}}\|^2,
\end{aligned} \tag{23}$$

by Lemma 2, where

$$b_n = C\bar{g}\sqrt{s}\left(\sum_k \sqrt{\frac{r}{r_k}}\frac{\sigma_{max,k}}{\sigma^2_{min,k}}\right) + C(\bar{g}+1)\sqrt{r}\left(1 + \sum_k \frac{\|\boldsymbol{\mathcal{A}}\|_F\sigma_{max,k}}{\sigma^2_{min,k}}\right),$$

and

$$d_n = C\bar{g}\sqrt{s}\left(\sum_k \sqrt{\frac{r}{r_k}\frac{1}{\sigma^2_{min,k}}}\right) + C(\bar{g}+1)\sqrt{r}\left(\sum_k \frac{\|\boldsymbol{\mathcal{A}}\|_F}{\sigma^2_{min,k}}\right).$$

Let $\mathbf{\Omega}_2 = \{\mathbf{\Delta} : \|\mathbf{\Delta}\|_1 \le b_n\|\mathbf{\Delta}\| + d_n\|\mathbf{\Delta}\|^2\}$. Assume $\|\widehat{\mathbf{\Delta}}\| = t$. This means

$$\inf_{\substack{\mathbf{\Delta}\in\mathbf{\Omega}_2 \\ \|\mathbf{\Delta}\|=t}} Q_n(\mathbf{U}\mathbf{g} + \mathbf{\Delta}) + \lambda\|\mathbf{U} + \mathbf{\Delta}_{\mathbf{U}}\|_1 - \lambda\|\mathbf{U}\|_1 < 0. \tag{24}$$

Lemma 1 shows that with probability at least $1 - (p \vee n)^{-C}$,

$$\sup_{\substack{\mathbf{\Delta}\in\mathbf{\Omega}_2 \\ \|\mathbf{\Delta}\|=t}} |Q_n(\mathbf{U}\mathbf{g} + \mathbf{\Delta}) - EQ_n(\mathbf{U}\mathbf{g} + \mathbf{\Delta})| \le C(b_n t + d_n t^2)\sqrt{\frac{\log(p \vee n)}{n}}. \tag{25}$$

(24) and (25) together means that there exists $\mathbf{\Delta}$ with$\|\mathbf{\Delta}\|=t$ such that

$$\begin{aligned} EQ_n(\mathbf{Ug}+\mathbf{\Delta}) &\leq \lambda\|\mathbf{U}\|_1-\lambda\|\mathbf{U}+\mathbf{\Delta}_\mathbf{U}\|_1+C(b_nt+d_nt^2)\sqrt{\frac{\log(p\vee n)}{n}} \\ &\leq \lambda\|(\widehat{\mathbf{\Delta}}_\mathbf{U})_S\|_1-\lambda\|(\widehat{\mathbf{\Delta}}_\mathbf{U})_{S^c}\|_1+C(b_nt+d_nt^2)\sqrt{\frac{\log(p\vee n)}{n}} \\ &\leq \lambda\|(\widehat{\mathbf{\Delta}}_\mathbf{U})_S\|_1+C\frac{b_nt+d_nt^2}{\sqrt{n}}\leq C\frac{\lambda}{\bar{g}}(b_nt+d_nt^2)+C(b_nt+d_nt^2)\sqrt{\frac{\log(p\vee n)}{n}}, \end{aligned}$$

where the second inequality uses the same arguments that leads to (22) starting from (21), and the last step used (23).

Using assumption C8, we then have

$$c_1(t^2\wedge t)\leq C\frac{\lambda}{\bar{g}}(b_nt+d_nt^2)+C(b_nt+d_nt^2)\sqrt{\frac{\log(p\vee n)}{n}},$$

which implies $t\leq Cb_n\lambda/\bar{g}$ if $d_n\lambda/\bar{g}=o(1)$. □

Remark 3 *To incorporate the intercept μ, let $\widehat{\delta}_\mu=\widehat{\mu}-\mu$. Then using similar arguments with minor modifications, we have $(\widehat{\delta}_\mu,\widehat{\mathbf{\Delta}})\in\mathbf{\Omega}:=\{(\delta_\mu,\mathbf{\Delta}):\|(\mathbf{\Delta}_\mathbf{U})_{S^c}\|_1\leq 3\|(\mathbf{\Delta}_\mathbf{U})_S\|_1+\|\mathbf{\Delta}_\mathbf{g}\|_1+|\delta_\mu|\}\}$, and (23) is replaced by $\|\widehat{\mathbf{\Delta}}\|_1\leq b_n\|\widehat{\mathbf{\Delta}}\|+d_n\|\widehat{\mathbf{\Delta}}\|^2+\bar{g}|\widehat{\delta}_\mu|$. The rest of the proof still can proceed as before using $(\delta_\mu,\mathbf{\Delta})\in\mathbf{\Omega}_2:=\{(\delta_\mu,\mathbf{\Delta}):\|\mathbf{\Delta}\|_1\leq b_n\|\mathbf{\Delta}\|+d_n\|\mathbf{\Delta}\|^2+\bar{g}|\delta_\mu|\}$ and $\|\mathbf{\Delta}\|+|\delta_\mu|=t$ instead of $\|\mathbf{\Delta}\|=t$. Then we can see that the bound $\|\widehat{\mathbf{\Delta}}\|+|\widehat{\boldsymbol{\delta}}_\mu|\leq b_n\lambda/\bar{g}$ holds.*

Regarding the roles of the following lemmas, Lemmas 1 and 2 are used in the proof of Theorem 2 while Lemma 3 is used in the proof of Lemma 1.

Lemma 1 *Let $\mathbf{\Omega}_2=\{\mathbf{\Delta}:\|\mathbf{\Delta}\|_1\leq b_n\|\mathbf{\Delta}\|+d_n\|\mathbf{\Delta}\|^2\}$ as defined in the proof of Theorem 2. With probability at least $1-(p\vee n)^{-C}$ for some constant $C>0$,*

$$\begin{aligned} &\sup_{\|\mathbf{\Delta}\|\in\mathbf{\Omega}_2,\|\mathbf{\Delta}\|\leq t}\Bigg|\frac{1}{n}\sum_{i=1}^n\rho_\tau\big(y_i-\boldsymbol{x}_i^T(\boldsymbol{U}\boldsymbol{g}+\mathbf{\Delta})\big)-\frac{1}{n}\sum_{i=1}^n\rho_\tau\big(y_i-\boldsymbol{x}_i^T\boldsymbol{U}\boldsymbol{g}\big) \\ &\qquad -E\rho_\tau\big(y_i-\boldsymbol{x}_i^T(\boldsymbol{U}\boldsymbol{g}+\mathbf{\Delta})\big)+E\rho_\tau\big(y_i-\boldsymbol{x}_i^T\boldsymbol{U}\boldsymbol{g}\big)\Bigg| \\ \leq\ & C(b_nt+d_nt^2)\sqrt{\log(p\vee n)/n}. \end{aligned}$$

Proof. Let

$$A(t)=\sup_{\mathbf{\Delta}\in\mathbf{\Omega}_2,\|\mathbf{\Delta}\|\leq t}n^{-1/2}\Big|\mathbb{G}_n\big[\rho_\tau\big(y-\boldsymbol{x}^T(\boldsymbol{U}\boldsymbol{g}+\mathbf{\Delta})\big)-\rho_\tau\big(y-\boldsymbol{x}^T\boldsymbol{U}\boldsymbol{g}\big)\big]\Big|,$$

where $\mathbb{G}_nf(x_i)=\sqrt{n}(P_nf-Pf)$ is the empirical process. Note that for any $\mathbf{\Delta}$ with $\|\mathbf{\Delta}\|\leq t$, by the Lipschitz property of ρ_τ, we have

$$Var\big(\mathbb{G}_n\big[\rho_\tau\big(y-\boldsymbol{x}^T(\boldsymbol{U}\boldsymbol{g}+\mathbf{\Delta})\big)-\rho_\tau\big(y-\boldsymbol{x}^T\boldsymbol{U}\boldsymbol{g}\big)\big]\big)\leq E(\boldsymbol{x}^T\mathbf{\Delta})^2\leq t^2/\underline{f}.$$

Then an application of Lemma 2.3.7 of van der Vaart and Wellner (1996) yields

$$P(A(t) \geq M) \leq \frac{2P(B(t) \geq \frac{M}{4})}{1 - t^2/(nM^2 \underline{f})}, \tag{26}$$

if the denominator above is positive, where

$$B(t) = \sup_{\boldsymbol{\Delta} \in \boldsymbol{\Omega}_2, \|\boldsymbol{\Delta}\| \leq t} n^{-1/2} \Big| \mathbb{G}_n^s \big[\rho_\tau \big(y - \boldsymbol{x}^T (\boldsymbol{U}\boldsymbol{g} + \boldsymbol{\Delta}) \big) - \rho_\tau \big(y - \boldsymbol{x}^T \boldsymbol{U}\boldsymbol{g} \big) \big] \Big|,$$

$\mathbb{G}_n^s f(\mathbf{x}, y) = n^{-1/2} \sum_i \delta_i f(\mathbf{x}_i, y_i)$, and $\delta_i \in \{-1, 1\}$ are independent Rademacher variables.

We also have

$$\begin{aligned}
P(B(t) \geq M) &\leq e^{-\gamma M} E e^{\gamma B(t)} \\
&\leq e^{-\gamma M} E \left[\exp \left\{ 2\gamma \sup_{\boldsymbol{\Delta} \in \boldsymbol{\Omega}_2, \|\boldsymbol{\Delta}\| \leq t} (1/n) \Big| \sum_i \delta_i \mathbf{x}_i^T \boldsymbol{\Delta} \Big| \right\} \right] \\
&\leq e^{-\gamma M} E \left[\exp \left\{ 2\gamma \sup_{\boldsymbol{\Delta} \in \boldsymbol{\Omega}_2, \|\boldsymbol{\Delta}\| \leq t} \Big\| \frac{1}{n} \sum_i \mathbf{x}_i \delta_i \Big\|_\infty \|\boldsymbol{\Delta}\|_1 \right\} \right] \\
&\leq C p e^{-\gamma M} \exp\{ C \gamma^2 (b_n t + d_n t^2)^2 / n \} \\
&\leq p \exp \left\{ -C \frac{n M^2}{(b_n t + d_n t^2)^2} \right\},
\end{aligned}$$

where the 1st line uses Markov's inequality, the 2nd uses the contraction property of the Rademacher process (see Theorem 2.3 of Koltchinskii, 2011), the 3rd uses the simple inequality $|\mathbf{a}^T \mathbf{b}| \leq \|\mathbf{a}\|_\infty \|\mathbf{b}\|_1$ for two vectors $\mathbf{a}$ and $\mathbf{b}$, the 4th uses Lemma 3 together with the fact that $\|\boldsymbol{\Delta}\|_1 \leq b_n t + d_n t^2$, and the last line is obtained by setting $\gamma \asymp nM/(b_n t + d_n t^2)^2$. Finally, taking $M \asymp (b_n t + d_n t^2)\sqrt{\log(p \vee n)/n}$ proves the lemma. ■

Lemma 2 *There exists $r_k \times r_k$ orthogonal matrix $\widehat{\mathbf{O}}_k$, $k = 1, \ldots, K$, such that*

$$\begin{aligned}
& \|(\widehat{\mathbf{U}}_K \widehat{\mathbf{O}}_K) \otimes \cdots \otimes (\widehat{\mathbf{U}}_1 \widehat{\mathbf{O}}_1) - \mathbf{U}_K \otimes \cdots \otimes \mathbf{U}_1 \|_F \\
\leq\ & C \left(\sum_k \sqrt{\frac{r}{r_k}} \frac{\sigma_{max,k}}{\sigma_{min,k}^2} \right) \|\widehat{\boldsymbol{\mathcal{A}}} - \boldsymbol{\mathcal{A}}\|_F + C \left(\sum_k \sqrt{\frac{r}{r_k}} \frac{1}{\sigma_{min,k}^2} \right) \|\widehat{\boldsymbol{\mathcal{A}}} - \boldsymbol{\mathcal{A}}\|_F^2,
\end{aligned}$$

and

$$\|\widehat{\boldsymbol{\mathcal{G}}} \times_1 \widehat{\mathbf{O}}_1^T \cdots \times_K \widehat{\mathbf{O}}_K^T - \boldsymbol{\mathcal{G}}\|_F \leq C \left(1 + \sum_k \frac{\|\boldsymbol{\mathcal{A}}\|_F \sigma_{max,k}}{\sigma_{min,k}^2} \right) \|\widehat{\boldsymbol{\mathcal{A}}} - \boldsymbol{\mathcal{A}}\|_F + C \left(\sum_k \frac{\|\boldsymbol{\mathcal{A}}\|_F}{\sigma_{min,k}^2} \right) \|\widehat{\boldsymbol{\mathcal{A}}} - \boldsymbol{\mathcal{A}}\|_F^2.$$

Proof. Since $\widehat{\mathbf{U}}_k$ and $\mathbf{U}_k$ are the left singular vectors of $\widehat{\boldsymbol{\mathcal{A}}}_{(k)}$ and $\boldsymbol{\mathcal{A}}_{(k)}$, respectively, by the Davis-Kahn theorem as stated in Theorem 3 of Yu et al. (2015), we have

$$\|\widehat{\mathbf{U}}_k \widehat{\mathbf{O}}_k - \mathbf{U}_k\|_F \leq C \frac{(\sigma_{max,k} + \|\widehat{\boldsymbol{\mathcal{A}}} - \boldsymbol{\mathcal{A}}\|_F) \|\widehat{\boldsymbol{\mathcal{A}}} - \boldsymbol{\mathcal{A}}\|_F}{\sigma_{min,k}^2},$$

for some orthogonal matrix $\widehat{\mathbf{O}}_k$. Note that $\widehat{\boldsymbol{\mathcal{A}}} = \widehat{\boldsymbol{\mathcal{G}}} \times_1 \widehat{\mathbf{U}}_1 \cdots \times_K \widehat{\mathbf{U}}_K = (\widehat{\boldsymbol{\mathcal{G}}} \times_1 \widehat{\mathbf{O}}_1^T \cdots \times_K \widehat{\mathbf{O}}_K^T) \times_1 \widehat{\mathbf{U}}_1\widehat{\mathbf{O}}_1 \cdots \times_K \widehat{\mathbf{U}}_K\widehat{\mathbf{O}}_K$. For simplicity of notation, in the following we denote $\widehat{\mathbf{U}}_k\widehat{\mathbf{O}}_k$ simply by $\widehat{\mathbf{U}}_k$ and $\widehat{\boldsymbol{\mathcal{G}}} \times_1 \widehat{\mathbf{O}}_1^T \cdots \times_K \widehat{\mathbf{O}}_K^T$ by $\widehat{\boldsymbol{\mathcal{G}}}$. Then

$$\begin{aligned}
& \|\widehat{\mathbf{U}}_K \otimes \cdots \otimes \widehat{\mathbf{U}}_1 - \mathbf{U}_K \otimes \cdots \otimes \mathbf{U}_1\|_F \\
\leq\ & \|\widehat{\mathbf{U}}_K \otimes \cdots \otimes \widehat{\mathbf{U}}_1 - \widehat{\mathbf{U}}_K \otimes \cdots \widehat{\mathbf{U}}_2 \otimes \mathbf{U}_1\|_F + \cdots + \|\widehat{\mathbf{U}}_K \otimes \mathbf{U}_{K-1} \cdots \otimes \mathbf{U}_1 - \mathbf{U}_K \otimes \cdots \mathbf{U}_2 \otimes \mathbf{U}_1\|_F \\
=\ & \sum_k \sqrt{\frac{r}{r_k}} \|\widehat{\mathbf{U}}_k - \mathbf{U}_k\|_F \\
\leq\ & C\left(\sum_k \sqrt{\frac{r}{r_k}} \frac{\sigma_{max,k}}{\sigma_{min,k}^2}\right) \|\widehat{\boldsymbol{\mathcal{A}}} - \boldsymbol{\mathcal{A}}\|_F + C\left(\sum_k \sqrt{\frac{r}{r_k}} \frac{1}{\sigma_{min,k}^2}\right) \|\widehat{\boldsymbol{\mathcal{A}}} - \boldsymbol{\mathcal{A}}\|_F^2, \qquad (27)
\end{aligned}$$

where the equality is due to $\|\mathbf{A} \otimes \mathbf{B}\|_F = \|\mathbf{A}\|_F\|\mathbf{B}\|_F$.

Furthermore,

$$\begin{aligned}
& \|\widehat{\boldsymbol{\mathcal{G}}} - \boldsymbol{\mathcal{G}}\|_F = \|(\widehat{\mathbf{U}}_K \otimes \cdots \otimes \widehat{\mathbf{U}}_1)^T \text{vec}(\widehat{\boldsymbol{\mathcal{A}}}) - \mathbf{U}_K \otimes \cdots \otimes \mathbf{U}_1)^T \text{vec}(\boldsymbol{\mathcal{A}})\|_F \\
\leq\ & \|(\widehat{\mathbf{U}}_K \otimes \cdots \otimes \widehat{\mathbf{U}}_1)^T (\text{vec}(\widehat{\boldsymbol{\mathcal{A}}}) - \text{vec}(\boldsymbol{\mathcal{A}}))\|_F + \|(\widehat{\mathbf{U}}_K \otimes \cdots \otimes \widehat{\mathbf{U}}_1 - \mathbf{U}_K \otimes \cdots \otimes \mathbf{U}_1)^T \text{vec}(\boldsymbol{\mathcal{A}})\|_F \\
\leq\ & \|\widehat{\mathbf{U}}_K \otimes \cdots \otimes \widehat{\mathbf{U}}_1\|_{op} \|\widehat{\boldsymbol{\mathcal{A}}} - \boldsymbol{\mathcal{A}}\|_F + \|\widehat{\mathbf{U}}_K \otimes \cdots \otimes \widehat{\mathbf{U}}_1 - \mathbf{U}_K \otimes \cdots \otimes \mathbf{U}_1\|_{op} \|\boldsymbol{\mathcal{A}}\|_F \\
\leq\ & C\left(1 + \sum_k \frac{\|\boldsymbol{\mathcal{A}}\|_F \sigma_{max,k}}{\sigma_{min,k}^2}\right) \|\widehat{\boldsymbol{\mathcal{A}}} - \boldsymbol{\mathcal{A}}\|_F + C\left(\sum_k \frac{\|\boldsymbol{\mathcal{A}}\|_F}{\sigma_{min,k}^2}\right) \|\widehat{\boldsymbol{\mathcal{A}}} - \boldsymbol{\mathcal{A}}\|_F^2, \qquad (28)
\end{aligned}$$

using that

$$\begin{aligned}
& \|\widehat{\mathbf{U}}_K \otimes \cdots \otimes \widehat{\mathbf{U}}_1 - \mathbf{U}_K \otimes \cdots \otimes \mathbf{U}_1\|_{op} \\
\leq\ & \|\widehat{\mathbf{U}}_K \otimes \cdots \otimes \widehat{\mathbf{U}}_1 - \widehat{\mathbf{U}}_K \otimes \cdots \widehat{\mathbf{U}}_2 \otimes \mathbf{U}_1\|_{op} + \cdots + \|\widehat{\mathbf{U}}_K \otimes \mathbf{U}_{K-1} \cdots \otimes \mathbf{U}_1 - \mathbf{U}_K \otimes \cdots \mathbf{U}_2 \otimes \mathbf{U}_1\|_{op} \\
=\ & \sum_k \|\widehat{\mathbf{U}}_k - \mathbf{U}_k\|_{op} \\
\leq\ & C\left(\sum_k \frac{\sigma_{max,k}}{\sigma_{min,k}^2}\right) \|\widehat{\boldsymbol{\mathcal{A}}} - \boldsymbol{\mathcal{A}}\|_F + C\left(\sum_k \frac{1}{\sigma_{min,k}^2}\right) \|\widehat{\boldsymbol{\mathcal{A}}} - \boldsymbol{\mathcal{A}}\|_F^2.
\end{aligned}$$

■

Lemma 3 *For any constant $\gamma > 0$,*

$$E[\exp\{\gamma \max_{1 \leq j \leq p} |\sum_i x_{ij}\delta_i|\}] \leq 2pe^{Cn\gamma^2},$$

where $\delta_i \in \{-1, 1\}$ are independent Rademacher variables.

Proof. We have

$$\begin{aligned}
& E[\exp\{\gamma \max_j |\sum_i x_{ij}\delta_i|\}] \\
=\ & E[\max_j \exp\{\gamma|\sum_i x_{ij}\delta_i|\}] \\
\leq\ & p \max_j E[\exp\{\gamma|\sum_i x_{ij}\delta_i|\}] \\
\leq\ & 2p \max_j E[\exp\{\gamma(\sum_i x_{ij}\delta_i)\}],
\end{aligned}$$

where the last step used the fact that for any symmetric random variable z, $E[e^{|z|}] \leq e[e^z + e^{-z}] = 2E[e^z]$. Using x_{ij} is sub-Gaussian and thus $x_{ij}\delta_i$ is also sub-Gaussian, we get $E[\exp\{\gamma(\sum_i x_{ij}\delta_i)\}] = (e^{C\gamma^2})^n$ which proved the lemma. ■

References

Animashree Anandkumar, Rong Ge, Daniel Hsu, Sham M. Kakade, and Matus Telgarsky. Tensor decompositions for learning latent variable models. *Journal of Machine Learning Research*, 15(80):2773–2832, 2014a.

Animashree Anandkumar, Rong Ge, and Majid Janzamin. Guaranteed non-orthogonal tensor decomposition via alternating rank-1 updates. *arXiv preprint arXiv:1402.5180*, 2014b.

Alexandre Belloni and Victor Chernozhukov. l1 -penalized quantile regression in high-dimensional sparse models. *The Annals of Statistics*, 39(1):82–130, 2011.

Xingwei Cao and Guillaume Rabusseau. Tensor regression networks with various low-rank tensor approximations. *arXiv preprint arXiv:1712.09520*, 2017.

Daniele Castellana and Davide Bacciu. Tensor decompositions in recursive neural networks for tree-structured data. *arXiv preprint arXiv:2006.10619*, 2020.

Antoni B. Chan, Mulloy Morrow, and Nuno Vasconcelos. Analysis of crowded scenes using holistic properties. In *IEEE Intl. Workshop on Performance Evaluation of Tracking and Surveillance (PETS 2009)*, 2009.

Eric C. Chi and Tamara G. Kolda. On tensors, sparsity, and nonnegative factorizations. *SIAM Journal on Matrix Analysis and Applications*, 33(4):1272–1299, 2012.

James Ferryman and Ali Shahrokni. Pets2009: Dataset and challenge. In *IEEE International Workshop on Performance Evaluation of Tracking and Surveillance*, pages 1–6, 2009.

Weiwei Guo, Irene Kotsia, and Ioannis Patras. Tensor learning for regression. *IEEE Transactions on Image Processing*, 21(2):816–827, 2012.

Johan Håstad. Tensor rank is NP-complete. *Algorithms*, 11(4):644–654, 1990.

Kejun Huang, Nicholas D. Sidiropoulos, and Athanasios P. Liavas. A flexible and efficient algorithmic framework for constrained matrix and tensor factorization. *IEEE Transactions on Signal Processing*, 64(19):5052–5065, 2016.

Roger Koenker. *Quantile Regression.* Cambridge University Press, New York, 2005.

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

Arinbjörn Kolbeinsson, Jean Kossaifi, Yannis Panagakis, Adrian Bulat, Anima Anandkumar, Ioanna Tzoulaki, and Paul Matthews. Robust deep networks with randomized tensor regression layers. *arXiv preprint arXiv:1902.10758*, 2019.

Tamara G. Kolda and Brett W. Bader. Tensor decompositions and applications. *SIAM Review*, 51(3):455–500, 2009.

Vladimir Koltchinskii. *Oracle Inequalities in Empirical Risk Minimization and Sparse Recovery Problems*. Springer, New York, 2011.

Jean Kossaifi, Yannis Panagakis, Anima Anandkumar, and Maja Pantic. TensorLy: Tensor Learning in Python. *Journal of Machine Learning Research*, 20(1):1–6, 2019.

Jean Kossaifi, Zachary C Lipton, Arinbjorn Kolbeinsson, Aran Khanna, Tommaso Furlanello, and Anima Anandkumar. Tensor regression networks. *Journal of Machine Learning Research*, 21:1–21, 2020.

Lieven De Lathauwer, Bart De Moor, and Joos Vandewalle. A multilinear singular value decomposition. *SIAM Journal on Matrix Analysis and Applications*, 21(4):1253–1278, 2000.

Lexin Li and Xin Zhang. Parsimonious tensor response regression. *Journal of the American Statistical Association*, 112(519):1131–1146, 2017.

Xiaoshan Li, Da Xu, Hua Zhou, and Lexin Li. Tucker tensor regression and neuroimaging analysis. *Statistics in Biosciences*, 10:520–545, 2018.

Ji Liu, Jun Liu, Peter Wonka, and Jieping Ye. Sparse non-negative tensor factorization using columnwise coordinate descent. *Pattern Recognition*, 45(1):649–656, 2012.

Ji Liu, Przemyslaw Musialski, Peter Wonka, and Jieping Ye. Tensor completion for estimating missing values in visual data. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 35:208–220, 2013.

Yipeng Liu, Jiani Liu, and Ce Zhu. Low-rank tensor train coefficient array estimation for tensor-on-tensor regression. *IEEE Transactions on Neural Networks and Learning Systems*, to appear, 2020.

Nicolai Meinshausen. Quantile regression forests. *Journal of Machine Learning Research*, 7:983–999, 2006.

Alexander Novikov, Dmitrii Podoprikhin, Anton Osokin, and Dmitry P Vetrov. Tensorizing neural networks. In *International Conference on Neural Information Processing Systems (NIPS)*, pages 442–450, 2015.

I. V. Oseledets. Tensor-train decomposition. *SIAM Journal on Scientific Computing*, 33 (5):2295–23, 2011.

Guillaume Rabusseau and Hachem Kadri. Low-rank regression with tensor responses. In *International Conference on Neural Information Processing Systems (NIPS)*, pages 1875–1883, 2016.

Garvesh Raskutti, Ming Yuan, and Han Chen. Convex regularization for high-dimensional multiresponse tensor regression. *The Annals of Statistics*, 47(3):1554–1584, 06 2019.

Steffen Rendle and Lars Schmidt-Thieme. Pairwise interaction tensor factorization for personalized tag recommendation. In *ACM International Conference on Web Search and Data Mining*, pages 81 – 90, 2010.

Alexander Shapiro. Asymptotic theory of overparameterized structural models. *Journal of the American Statistical Association*, 81(393):142–149, 1986.

Shaden Smith, Alec Beri, and George Karypis. Constrained tensor factorization with accelerated ao-admm. In *International Conference on Parallel Processing (ICPP)*, pages 111–120, 2017.

Jiahao Su, Wonmin Byeon, Furong Huang, Jan Kautz, and Animashree Anandkumar. Convolutional tensor-train lstm for spatio-temporal learning. *arXiv preprint arXiv:2002.09131*, 2020.

Will Wei Sun and Lexin Li. STORE: sparse tensor response regression and neuroimaging analysis. *Journal of Machine Learning Research*, 18(1):1–37, 2017.

Will Wei Sun and Lexin Li. Dynamic tensor clustering. *Journal of the American Statistical Association*, 114(528):1894–1907, 2019.

Will Wei Sun, Junwei Lu, Han Liu, and Guang Cheng. Provable sparse tensor decomposition. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 79(3): 899–916, 2017.

James W. Taylor. A quantile regression neural network approach to estimating the conditional density of multiperiod returns. *Journal of Forecasting*, 19(4):299–311, 2000.

Madeleine Udell, Corinne Horn, Reza Zadeh, and Stephen Boyd. Generalized low rank models. *Foundations and Trends® in Machine Learning*, 9:1–118, 2016.

Aad W. van der Vaart. *Asymptotic Statistics*. Cambridge University Press, Cambridge, 1998.

Aad W. van der Vaart and Jon A. Wellner. *Weak Convergence and Empirical Processes*. Springer, New York, 1996.

Di Wang, Heng Lian, Yao Zheng, and Guodong Li. High-dimensional vector autoregressive time series modeling via tensor decomposition. *Available at http://dx.doi.org/10.2139/ssrn.3453802*, 2019.

Huixia Wang, Zhongyi Zhu, and Jianhui Zhou. Quantile regression in partially linear varying coefficient models. *The Annals of Statistics*, 37(6B):3841–3866, 2009.

Qi Xie, Qian Zhao, Deyu Meng, , and Zongben Xu. Kronecker-basis-representation based tensor sparsity and its applications to tensor recovery. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 40(8):1888–1902, 2018.

Congrui Yi and Jian Huang. Semismooth newton coordinate descent algorithm for elastic-net penalized huber loss regression and quantile regression. *Journal of Computational and Graphical Statistics*, 26(3):547–557, 2017.

Liqun Yu, Nan Lin, and Lan Wang. A parallel algorithm for large-scale nonconvex penalized quantile regression. *Journal of Computational and Graphical Statistics*, 26(4):935–939, 2017.

Yi Yu, Tengyao Wang, and Richard J. Samworth. A useful variant of the Davis–Kahan theorem for statisticians. *Biometrika*, 102(2):315–323, 2015.

Anru Zhang. Cross: Efficient low-rank tensor completion. *Annals of Statistics*, 47(2): 936–964, 04 2019.

Zemin Zhang, Gregory Ely, Shuchin Aeron, Ning Hao, and Misha Kilmer. Novel methods for multilinear data completion and de-noising based on tensor-svd. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2014.

Bingyin Zhou, Biao Song, Mohammad Mehedi Hassan, and Atif Alamri. Multilinear rank support tensor machine for crowd density estimation. *Engineering Applications of Artificial Intelligence*, 72:382 – 392, 2018.

Hua Zhou, Lexin Li, and Hongtu Zhu. Tensor regression with applications in neuroimaging data analysis. *Journal of the American Statistical Association*, 108(502):540–552, 2013.