Computational Statistics and Data Analysis 199 (2024) 107998

Contents lists available at ScienceDirect

Computational Statistics and Data Analysis

journal homepage: www.elsevier.com/locate/csda

Conditional mean dimension reduction for tensor time series

Chung Eun Lee [a,*,1], Xin Zhang [b]

[a] Paul H. Chook Department of Information Systems and Statistics, Baruch College, New York, 10010, NY, United States of America
[b] Department of Statistics, Florida State University, Tallahassee, 213456, FL, United States of America

ARTICLE INFO

Keywords:
Dimension reduction
Factor model
Martingale difference divergence
Nonlinearity
Tensor decomposition
Tensor time series

ABSTRACT

The dimension reduction problem for a stationary tensor time series is addressed. The goal is to remove linear combinations of the tensor time series that are mean independent of the past, without imposing any parametric models or distributional assumptions. To achieve this goal, a new metric called cumulative tensor martingale difference divergence is introduced and its theoretical properties are studied. Unlike existing methods, the proposed approach achieves dimension reduction by estimating a distinctive subspace that can fully retain the conditional mean information. By focusing on the conditional mean, the proposed dimension reduction method is potentially more accurate in prediction. The method can be viewed as a factor model-based approach that extends the existing techniques for estimating central subspace or central mean subspace in vector time series. The effectiveness of the proposed method is illustrated by extensive simulations and two real-world data applications.

1. Introduction

Tensor data is nowadays prevalent in numerous applications, including tensor time series, where the observation is tensor-valued (i.e., a multidimensional array) at each time point. In general, tensor time series data has a complex structure, meaningful temporal dependence, and requires dimension reduction for efficient analysis.

A primary problem in tensor time series analysis is reducing the dimension while retaining the information of interest and the tensor structure. Traditionally, the dimension reduction methods in time series analysis were developed for the multivariate time series data; see Lam et al. (2011); Matteson and Tsay (2011); Lam and Yao (2012); Lee and Shao (2018), among others. Recently, there are methods that particularly focus on the dimension reduction for matrix or tensor time series data, including the factor model for matrix time series by Wang et al. (2019), the constrained factor models for matrix time series by Chen et al. (2020), the factor models for tensor time series by Chen et al. (2021), the two-way transformed factor model for matrix-variate time series by Gao and Tsay (2021), α-PCA method for matrix-variate time series by Chen and Fan (2021). However, all these above-mentioned methods adopt a linear metric, a covariance matrix, that can only summarize the linear dependence to achieve the dimension reduction. Thus, these methods target to find linear subspaces where the transformed matrix or tensor times series has a strong linear dynamic structure. If the data is Gaussian, the linear metric can detect the full dependence and the existing methods shall achieve an accurate dimension reduction. However, if the data is not Gaussian or the data has nonlinear dependence, the linear metric may fail to summarize the full dynamic.

* Corresponding author.
E-mail addresses: chungeun.lee@baruch.cuny.edu (C.E. Lee), henry@stat.fsu.edu (X. Zhang).
[1] 55 Lexington Ave, New York, NY, 10010, 646-312-3378.

https://doi.org/10.1016/j.csda.2024.107998
Received 3 October 2023; Received in revised form 28 May 2024; Accepted 31 May 2024

Available online 11 June 2024

In the tensor regression context, where independent and identically distributed data are assumed, the dimension reduction techniques have been extensively studied. On one hand, there are methods reduce the dimensionality of a tensor predictor by constructing dimension-folding subspaces; see, for example, Li et al. (2010), Ding and Cook (2014), Sheng and Yuan (2020), Wang et al. (2022). On the other hand, Rabusseau and Kadri (2016), Li and Zhang (2017), Sun and Li (2017), Lee et al. (2023) incorporated dimension reduction and low-rank decomposition techniques in tensor response regression models.

In this article, we propose a dimension reduction procedure of a stationary tensor times series, $\{\mathcal{X}_t\}_{t=1}^n$, $\mathcal{X}_t \in \mathbb{R}^{r_1 \times \cdots \times r_m}$, especially focusing on the conditional mean of a tensor time series given the past information, $\mathcal{F}_{t-1} = \sigma(\mathcal{X}_{t-1}, \mathcal{X}_{t-2}, \cdots)$, which is the main interest of modeling the behavior of the data. More specifically, our goal is to seek subspaces that reconstruct the tensor time series into two parts: one part that contains the conditional mean information and the other part that is mean independent of the past. In other words, our approach can effectively reduce the number of parameters while preserving both the tensor structure and the conditional mean information without assuming a parametric time series model or a distributional assumption. Our proposal can be viewed as a factor model-based approach that extends the methods for estimating central subspace or central mean subspace in vector time series (Park et al., 2010, 2009) to tensor time series. These existing model-free time series dimension reduction methods are flexible but not scalable or directly applicable to large number variables, which is typical in tensor time series.

While the existing methods achieve dimension reduction by using the linear metric, our approach relies on the new metric, called the cumulative tensor martingale difference divergence (CTMDD), that can summarize the mean dependence and overcome some limitations of the covariance matrix-based approaches. Hence, we shall call our approach as the conditional mean dimension reduction for tensor time series. Since the new metric measures the mean dependence, it can gather the nonlinear dependence along with the linear dependence that appears in the conditional mean. Therefore, our targeted subspace indeed contains the subspace that the existing method seeks where two subspaces become equivalent for a particular case. Moreover, our approach can retrieve the conditional mean which is the optimal predictor in terms of the mean squared error. Thus, modeling the behavior of a tensor time series with our dimension reduction method could produce more accurate predictions. Compared to the existing methods, our approach is more robust and flexible to the dependence, and practically useful for forecasting. Similar to the existing methods, our approach has a tensor factor model representation which has a sophisticated difference compared to the existing tensor factor model.

Our new metric is built upon and extends the martingale difference divergence (MDD) metric. Shao and Zhang (2014) proposed the MDD for the variable screening purposes which can capture the mean dependence between a scalar and a vector. Furthermore, Lee and Shao (2018) and Lee and Shao (2020) extended the MDD and introduced the cumulative martingale difference divergence matrix (CMDDM) and the cumulative volatility martingale difference divergence matrix to achieve the dimension reduction for a stationary multivariate time series. However, they all handle the multivariate time series instead of the tensor time series. In order to achieve our goal, we particularly generalize the CMDDM of Lee and Shao (2018) in two ways. First, we extend the metric in Lee and Shao (2018) and define the CTMDD so that it is well defined for two tensors. Second, we develop the new metric for a general class of distances motivated by the recent progress of adopting new distances for metrics. This allows us to efficiently quantify the mean dependence for tensors which often have large dimensions; see Chakraborty and Zhang (2019) and Zhou and Zhu (2021). We further show that the generalized metric indeed fully quantifies the mean dependence between two tensors by using new techniques. Later, we propose our dimension reduction approach with the extended metric to estimate the number and the linear forms of the data that retain the conditional mean information. Our approach is computationally efficient and simple to implement regardless of the fact that we consider the tensor time series.

The rest of the article is organized as follows. In Section 2, we briefly review the key tensor notations, operations, and the CMDDM (Lee and Shao, 2018). Section 3 introduces the new metric CTMDD and we introduce our approach to reduce the dimension using the new metric. Section 4 presents numerical studies, and Section 5 presents applications of the proposed method to two real data sets. Section 6 concludes the paper with a short discussion. All proofs and additional simulations are relegated to the supplementary material.

2. Preparation

2.1. Notations

Let $\mathcal{A} \in \mathbb{R}^{r_1 \times \cdots \times r_m}$ be a m-th order tensor and $\mathcal{A}_{i_1,\cdots,i_m}$ be the $(i_1, \cdots, i_m)$ element of $\mathcal{A}$. The order of a tensor is the number of its modes. A tensor fiber is defined by fixing every index of the tensor but one. For instance, $\mathcal{A}_{:,j_2,\cdots,j_m} \in \mathbb{R}^{r_1}$ is a mode-1 fiber for given $j_2, \cdots, j_m$. The Frobenius norm of $\mathcal{A}$ is $\|\mathcal{A}\|_F^2 = \langle \mathcal{A}, \mathcal{A} \rangle_F = \sum_{i_1,\cdots,i_m} \mathcal{A}_{i_1,\cdots,i_m}^2$. The vec($\mathcal{A}$) operator stacks all the entries of a tensor into one column vector, so that an entry $\mathcal{A}_{i_1 \cdots i_m}$ becomes the j-th entry of vec($\mathcal{A}$), where $j = 1 + \sum_{k=1}^m (i_k - 1) \prod_{k'=1}^{k-1} r_{k'}$. The mode-$k$ matricization, $\mathcal{A}_{(k)}$, transfers a tensor $\mathcal{A}$ into a matrix, denoted by $\mathcal{A}_{(k)}$, so that the $(i_1, \cdots, i_m)$ element of $\mathcal{A}$ is the (i_k, j) element of the matrix $\mathcal{A}_{(k)}$, where $j = 1 + \sum_{k' \neq k} (i_{k'} - 1) \prod_{k'' < k', k'' \neq k} r_{k''}$. The k-mode product of a tensor $\mathcal{A}$ and a matrix $\mathbf{C} \in \mathbb{R}^{s \times r_k}$ is a m-th order tensor denoted as $\mathcal{A} \times_{(k)} \mathbf{C} \in \mathbb{R}^{r_1 \times \cdots \times r_{k-1} \times s \times r_{k+1} \times \cdots \times r_m}$, where each element is the product of mode-k fiber of $\mathcal{A}$ multiplied by $\mathbf{C}$. The Tucker decomposition of a tensor is defined as $\mathcal{A} = \mathbf{D} \times_{(1)} \mathbf{\Gamma}_1 \times_{(2)} \cdots \times_{(m)} \mathbf{\Gamma}_m$, where $\mathbf{D} \in \mathbb{R}^{d_1 \times \cdots \times d_m}$ is the core tensor, and $\mathbf{\Gamma}_k \in \mathbb{R}^{r_k \times d_k}$, $k = 1, \cdots, m$, are the factor matrices. It is a low rank decomposition of the original tensor $\mathcal{A}$. For convenience, we shall denote the Tucker decomposition as $[\![\mathbf{D}; \mathbf{\Gamma}_1, \cdots, \mathbf{\Gamma}_m]\!]$. We refer to Kolda and Bader (2009) for more background on tensor decompositions.

For a vector $X = (x_1, \cdots, x_p) \in \mathbb{R}^p$, the Euclidean norm of X is $\|X\| = \left(x_1^2 + \cdots + x_p^2\right)^{1/2}$. For a square matrix $\mathbf{A} = (\mathbf{A}_{i,j})_{i,j=1}^p$, we denote the spectral norm and the Frobenius norm of $\mathbf{A}$ as $\|\mathbf{A}\|$ and $\|\mathbf{A}\|_F$, respectively, where $\|\mathbf{A}\| = \sqrt{\lambda_{\max}(\mathbf{A}^\top \mathbf{A})}$ and $\lambda_{\max}(\mathbf{A}^\top \mathbf{A})$ is

the largest eigenvalue of $\mathbf{A}^\top\mathbf{A}$ and $\|\mathbf{A}\|_F = \sqrt{\sum_{i,j}\mathbf{A}_{i,j}^2}$. The trace of $\mathbf{A}$ is $\mathrm{tr}(\mathbf{A}) = \sum_{i=1}^p \mathbf{A}_{i,i}$ and we denote the Kronecker product as $\otimes$. The orthogonal complement of S is $S^\perp$, $\perp\!\!\!\perp$ denotes the independence, and $\mathbf{0}$ is a tensor with zero for all entries.

2.2. Cumulative martingale difference divergence matrix

We briefly review the cumulative martingale difference divergence matrix (Lee and Shao, 2018), which is related to our new metric introduced in the next section. Following their notation, we will denote $\mathbf{\Gamma}_{h_0}$ as the CMDDM. If a multivariate time series $X_t \in \mathbb{R}^p$ satisfies $\mathbb{E}(\|X_t\|^2) < \infty$, then

$$\mathbf{\Gamma}_{h_0} = \sum_{h=1}^{h_0} \left\{-\mathbb{E}\left[\{X_t - \mathbb{E}(X_t)\}\{X_t' - \mathbb{E}(X_t')\}^\top \|X_{t-h} - X_{t-h}'\|\right]\right\}, \tag{1}$$

where (X_t', X_{t-h}') is an independent copy of (X_t, X_{t-h}). From the definition of the matrix $\mathbf{\Gamma}_{h_0}$, it is a $p\times p$ real, symmetric, and positive semi-definite matrix. By a direct consequence of Theorem 1 in Lee and Shao (2018), we have the following result.

Lemma 2.1. *For a stationary multivariate time series X_t which satisfies $\mathbb{E}(\|X_t\|^2) < \infty$, we have*

1. *There exist $p-d$ linearly independent combinations of X_t such that they are mean independent of $\{X_{t-h}\}_{h=1}^{h_0}$ if and only if* $\mathrm{rank}\left(\mathbf{\Gamma}_{h_0}\right) = d$.
2. *Let $\alpha \in \mathrm{span}^\perp\left(\mathbf{\Gamma}_{h_0}\right)$, then $\mathbb{E}(\alpha^\top X_t \mid X_{t-h}) = \mathbb{E}(\alpha^\top X_t)$ a.s. for $h = 1, \cdots, h_0$.*

The above lemma indicates that the space spanned by $\mathbf{\Gamma}_{h_0}$ is closely related to the linear combinations of X_t that are mean dependent of $(X_{t-1}, \cdots, X_{t-h_0})$.

3. Conditional mean dimension reduction

In this section, we propose our approach to achieve the dimension reduction for $\mathcal{X}_t \in \mathbb{R}^{r_1\times\cdots\times r_m}$ considering the conditional mean and provide a theoretical justification. Moreover, there is a factor model representation for our approach and we shall first introduce, arguably more general, tensor factor model which has a subtle difference compared to the existing tensor factor model.

3.1. Factor model representation to conditional mean dimension reduction

Our motivation is to reduce the dimension without losing the conditional mean information so that the dimension reduction is achieved with the least amount of loss on the prediction accuracy. Notice that data can always be decomposed into two parts where one part is mean dependent on the past and the other part being mean independent, i.e.,

$$\mathcal{X}_t = \mathbb{E}(\mathcal{X}_t \mid \mathcal{F}_{t-1}) + \mathcal{E}_t = [\![\mathcal{Z}_t; \mathbf{C}_1, \cdots, \mathbf{C}_m]\!] + \mathcal{E}_t, \tag{2}$$

where $\mathcal{E}_t = \mathcal{X}_t - \mathbb{E}(\mathcal{X}_t \mid \mathcal{F}_{t-1}) \in \mathbb{R}^{r_1\times\cdots\times r_m}$ is a martingale difference sequence that is mean independent of the past, and $[\![\mathcal{Z}_t; \mathbf{C}_1, \cdots, \mathbf{C}_m]\!]$ is the mean dependent part driven by a latent factor time series $\mathcal{Z}_t \in \mathbb{R}^{d_1\times\cdots\times d_m}$, $d_k < r_k$, $k = 1, \cdots, m$, and factor loading matrices $\{\mathbf{C}_k\}_{k=1}^m$, $\mathbf{C}_k \in \mathbb{R}^{r_k\times d_k}$ that are semi-orthogonal, i.e., $\mathbf{C}_k^\top\mathbf{C}_k = I_{d_k}$.

Similar to the existing factor models, it is worth pointing out that $\mathcal{Z}_t$ and $\{\mathbf{C}_k\}_{k=1}^m$ are not unique since we can replace those by $\left([\![\mathcal{Z}_t; \mathbf{H}_1, \cdots, \mathbf{H}_m]\!], \{\mathbf{C}_k\mathbf{H}_k^\top\}_{k=1}^m\right)$ with orthogonal matrices $\{\mathbf{H}_k\}_{k=1}^m$, $\mathbf{H}_k \in \mathbb{R}^{d_k\times d_k}$ that produce the same data $\mathcal{X}_t$. However, the linear subspaces spanned by the columns of $\{\mathbf{C}_k\}_{k=1}^m$, denoted by $\{\mathrm{span}(\mathbf{C}_k)\}_{k=1}^m$, are unique and identifiable; see Section 3 in Chen et al. (2021) for more discussion. Therefore, our goal is to estimate the dimensions of the factor $(d_1, \cdots, d_m)$ and the identifiable subspaces, often called as factor loading spaces $\{\mathrm{span}(\mathbf{C}_k)\}_{k=1}^m$ that fully carry the conditional mean information. Let $(\mathbf{C}_k, \mathbf{C}_{k,0})$ be an orthogonal matrix for $k = 1, \cdots, m$. Under our factor model (2), we observe that

$$\mathcal{X}_t \times_{(k)} \mathbf{C}_{k,0}^\top = \mathcal{E}_t \times_{(k)} \mathbf{C}_{k,0}^\top,\ k = 1, \cdots, m.$$

This is further identical to

$$\mathbf{C}_{k,0}^\top(\mathcal{X}_t)_{(k)} = \mathbf{C}_{k,0}^\top(\mathcal{E}_t)_{(k)},\ k = 1, \cdots, m, \tag{3}$$

which implies $\mathbb{E}(\mathbf{C}_{k,0}^\top(\mathcal{X}_t)_{(k)} \mid \mathcal{F}_{t-1}) = \mathbb{E}(\mathbf{C}_{k,0}^\top(\mathcal{X}_t)_{(k)})$ *a.s.* Hence, our goal of estimating $(d_k)_{k=1}^m$ and $\{\mathbf{C}_k\}_{k=1}^m$ is equivalent to searching the number and the linear combinations of $(\mathcal{X}_t)_{(k)}$ such that the transformed series is mean independent of the past for each mode $k = 1, \cdots, m$.

Remark 3.1. Interestingly, our dimension reduction approach is related to the time series central subspace (TS-CS) (Park et al., 2010) for tensor time series. To demonstrate the connection, we shall briefly review TS-CS in Park et al. (2010). Park et al. (2010) consider a univariate time series $x_t \in \mathbb{R}$ and search for TS-CS which is a minimal subspace of $\mathrm{span}(\mathbf{\Phi})$, where $\mathbf{\Phi} \in \mathbb{R}^{p\times d}$, $d < p$ satisfies

$$x_t \perp\!\!\!\perp X_{t-1} \mid \mathbf{\Phi}^\top X_{t-1},$$

here $\perp\!\!\!\perp$ denotes the independence and $X_{t-1} = (x_{t-1}, \cdots, x_{t-p})^\top \in \mathbb{R}^p$. As described in Park et al. (2010), TS-CS may not exist; see Proposition 1 in Park et al. (2010) for more discussion. We extend their notion of TS-CS from vector time series to tensor time series. Specifically, we introduce the mode-k tensor TS-CS as follows.

Definition 3.1. The mode-k tensor time series central subspace is a minimal subspace of span($\widetilde{\mathbf{C}}_k$), where $\widetilde{\mathbf{C}}_k \in \mathbb{R}^{r_k \times d_k}$, $d_k < r_k$ satisfies

$$\mathcal{X}_t \perp\!\!\!\perp (\mathcal{X}_{t-1}, \mathcal{X}_{t-2}, \cdots) \mid \left(\mathcal{X}_{t-1} \times_{(k)} \widetilde{\mathbf{C}}_k^\top, \mathcal{X}_{t-2} \times_{(k)} \widetilde{\mathbf{C}}_k^\top, \cdots\right). \tag{4}$$

When the mode-k tensor TS-CS exists for each mode, the factor loading spaces span($\mathbf{C}_k$) contains or become equivalent to the mode-k tensor TS-CS if

$$\left(\mathbf{C}_{k,0}^\top (\mathcal{E}_{t-1})_{(k)}, \mathbf{C}_{k,0}^\top (\mathcal{E}_{t-2})_{(k)}, \cdots\right) \perp\!\!\!\perp \left(\mathcal{X}_t, \mathbf{C}_k^\top (\mathcal{X}_{t-1})_{(k)}, \mathbf{C}_k^\top (\mathcal{X}_{t-2})_{(k)}, \cdots\right), \tag{5}$$

for $k = 1, \cdots, m$. This is due to the fact that the factor loading matrix $\mathbf{C}_k$ satisfies (4) under the assumption in (5) and by the fact in (3) and Proposition 4.6 in Cook (1998). This further implies that the mode-k tensor TS-CS $\subseteq$ span($\mathbf{C}_k$), $k = 1, \cdots, m$, where the equality holds if dimensions of two subspaces are identical. Discussion based on the central subspace in times series can be extended to the central mean subspace in time series which is a minimal subspace of span($\mathbf{\Phi}$), where $\mathbf{\Phi} \in \mathbb{R}^{p \times \overline{d}}$, $\overline{d} < p$ satisfies

$$x_t \perp\!\!\!\perp \mathbb{E}(x_t \mid X_{t-1}) \mid \mathbf{\Phi}^\top X_{t-1}.$$

Similarly, we can generalize the time series central mean subspace in Park et al. (2009) to the mode-k tensor time series central mean subspace.

Definition 3.2. The mode-k tensor time series central mean subspace is a minimal subspace of span($\overline{\mathbf{C}}_k$), where $\overline{\mathbf{C}}_k \in \mathbb{R}^{r_k \times \overline{d}_k}$, $\overline{d}_k < r_k$ satisfies

$$\mathcal{X}_t \perp\!\!\!\perp \mathbb{E}\left(\mathcal{X}_t \mid \mathcal{X}_{t-1}, \mathcal{X}_{t-2}, \cdots\right) \mid \left(\mathcal{X}_{t-1} \times_{(k)} \overline{\mathbf{C}}_k^\top, \mathcal{X}_{t-2} \times_{(k)} \overline{\mathbf{C}}_k^\top, \cdots\right).$$

We find out that our factor loading space span($\mathbf{C}_k$) also contains the mode-k tensor time series central mean subspace. This can be easily verified by the natural connection that the mode-k tensor time series central mean subspace $\subseteq$ the mode-k tensor TS-CS $\subseteq$ span($\mathbf{C}_k$), $k = 1, \cdots, m$.

Remark 3.2. We further compare our factor model with the existing factor model in Wang et al. (2019) and Chen et al. (2021). We shall first review the existing factor model.

$$\mathcal{X}_t = [\![\mathcal{Y}_t; \mathbf{A}_1, \cdots, \mathbf{A}_m]\!] + \mathcal{W}_t, \tag{6}$$

where $\mathcal{X}_t$ is the observed tensor time series, $\mathcal{Y}_t \in \mathbb{R}^{v_1 \times \cdots \times v_m}$, $v_k < r_k$, $k = 1, \cdots, m$ is the latent factor series, $\mathcal{W}_t \in \mathbb{R}^{r_1 \times \cdots \times r_m}$ is white noise, i.e., $\mathrm{cov}(\mathcal{W}_t, \mathcal{W}_{t-h}) = \mathbf{0}$, $h \neq 0$, and $\{\mathbf{A}_k\}_{k=1}^m$, $\mathbf{A}_k \in \mathbb{R}^{r_k \times v_k}$ are the semi-orthogonal factor loading matrices assuming that the magnitude of $\mathcal{X}_t$ is taken into account in $\mathcal{Y}_t$. As mentioned in Chen et al. (2021), $\mathcal{Y}_t$ and $\{\mathbf{A}_k\}_{k=1}^m$ are not identifiable. Since the error series in (6) is white noise, it is obvious that $\mathcal{X}_t \times_{(k)} \mathbf{A}_{k,0}^\top = \mathcal{W}_t \times_{(k)} \mathbf{A}_{k,0}^\top$, $k = 1, \cdots, m$ is a white noise series, where $(\mathbf{A}_k, \mathbf{A}_{k,0})$ construct an orthogonal matrix. Therefore, the existing methods employ the linear metric, covariance matrix, to achieve the dimension reduction. This implies that the existing methods search for the linear transformations of the data that capture the full linear dependence.

The main difference between the tensor factor models in (6) and (2) is the error series. While the error in the existing tensor factor model is a white noise series, the error in our tensor factor model is a martingale difference sequence. With this subtle difference, our approach has several differences compared to the existing method. Under our factor model, we have a nice interpretation for the factors $\mathcal{Z}_t$ which produces the conditional mean $\mathbb{E}(\mathcal{X}_t \mid \mathcal{F}_{t-1})$ along with the factor loading matrices $\{\mathbf{C}_k\}_{k=1}^m$, whereas $[\![\mathcal{Y}_t; \mathbf{A}_1, \cdots, \mathbf{A}_m]\!]$ is not necessarily the conditional mean. This implies that our dimension reduction approach can fully recover the optimal predictor, the conditional mean, through the factors $\mathcal{Z}_t$ and the factor loading matrices $\{\mathbf{C}_k\}_{k=1}^m$ even after the dimension reduction is achieved. Due to this fact, our approach may allow us to have more accurate forecasting especially when building a model with our factors; see the real data applications in Section 5. Furthermore, the existing methods reduce the dimension by the linear dependence. Unless the data is Gaussian, the linear dependence is not sufficient to summarize the full dependence. On the other hand, our approach can fully carry the conditional mean information where both the linear and the nonlinear dependence may exist. Hence, our approach could be more robust and flexible to the dependence and the distribution of the data. Moreover, it is possible for our tensor factor model to detect more factors than the existing tensor factor model which may lead to a better prediction. In particular, we have the nested structure, i.e., span($\mathbf{A}_k$) $\subseteq$ span($\mathbf{C}_k$), $k = 1, \cdots, m$ with equality when the white noise $\mathcal{W}_t$ is a martingale difference sequence which is not true in general; see Example 2.2 and 2.3 in Shao (2011) for examples of a white noise that are not martingale difference.

We shall provide one example where two factor models target different factors and factor loading spaces to better understand the differences between two factor models.

Example 3.1. Let the data be

$$\mathcal{X}_t = [\![(\mathcal{Y}_{1,t}, \mathcal{Z}_{1,t}); \mathbf{M}_1, \mathbf{M}_2]\!] + \mathcal{V}_t,$$

where $\mathcal{V}_t \in \mathbb{R}^{r_1 \times r_2}$ is an i.i.d. mean zero error series, $\mathbf{M}_1 \in \mathbb{R}^{r_1 \times d_1}$, $\mathbf{M}_2 = (\widetilde{\mathbf{M}}_2, \overline{\mathbf{M}}_2)$, $\widetilde{\mathbf{M}}_2 \in \mathbb{R}^{r_2 \times v_2}$, $\overline{\mathbf{M}}_2 \in \mathbb{R}^{r_2 \times (d_2 - v_2)}$, $v_2 < d_2$, $\mathcal{Y}_{1,t} \in \mathbb{R}^{d_1 \times v_2}$ is not white noise, and $\mathcal{Z}_{1,t} \in \mathbb{R}^{d_1 \times (d_2 - v_2)}$ is white noise but not martingale difference.

Notice that $[\![\mathcal{Z}_{1,t}; \mathbf{M}_1, \overline{\mathbf{M}}_2]\!]$ is a white noise sequence along with $\mathcal{V}_t$. Then the error series in (6) becomes $\mathcal{W}_t = [\![\mathcal{Z}_{1,t}; \mathbf{M}_1, \overline{\mathbf{M}}_2]\!] + \mathcal{V}_t$ whereas the error series in (2) is identical to $\mathcal{E}_t = \mathcal{X}_t - [\![\mathbb{E}\left\{(\mathcal{Y}_{1,t}, \mathcal{Z}_{1,t}) \mid \mathcal{F}_{t-1}\right\}; \mathbf{M}_1, \mathbf{M}_2]\!]$. Due to this fact, two models have different factors and factor loading spaces. Under the factor model of the existing method in (6), the factor is $\mathcal{Y}_t = \mathcal{Y}_{1,t}$ and the factor loading spaces are $\left\{\text{span}(\mathbf{M}_1), \text{span}(\widetilde{\mathbf{M}}_2)\right\}$. On the other hand, under our factor model in (2), the factor becomes $\mathcal{Z}_t = \mathbb{E}\left\{(\mathcal{Y}_t, \mathcal{Z}_{1,t}) \mid \mathcal{F}_{t-1}\right\}$ and the factor loading spaces are $\left\{\text{span}(\mathbf{M}_1), \text{span}(\mathbf{M}_2)\right\} = \left\{\text{span}(\mathbf{M}_1), \text{span}(\widetilde{\mathbf{M}}_2, \overline{\mathbf{M}}_2)\right\}$, thus two approaches identify different factors and factor loading spaces. It would be interesting to observe the numerical performance of two approaches for several cases, e.g., when two target subspaces are identical or different. We shall address this question in our simulation. Lastly, we remark that our approach extends Lee et al. (2023) from i.i.d. tensor data to tensor time series. Lee et al. (2023) introduced a semiparametric tensor regression model for a tensor response and a vector predictor, while we introduce a factor model to achieve the dimension reduction for tensor time series. As we consider the temporal dependence and the dependence between two tensors, this extension is nontrivial and requires new techniques.

In the further sections, we shall introduce the estimation procedure with a new metric, the CTMDD, which allows us to estimate $\{\mathbf{C}_k\}_{k=1}^m$ indirectly by the fact in (3). Our approach consistently estimates $\{\mathbf{C}_k\}_{k=1}^m$ through the eigen-decomposition of the CTMDD, thus it does not require an iteration procedure and makes it simple to implement.

3.2. Cumulative tensor martingale difference divergence

For each mode k, our specific goal is to seek linear forms of $(\mathcal{X}_t)_{(k)}$ that are mean independent of the past $\mathcal{F}_{t-1}$. Since we have a finite number of observations, we approximate the mean independence of the linear transformation of $(\mathcal{X}_t)_{(k)}$ on $\mathcal{F}_{t-1}$ by considering $\mathcal{F}_{t-1,t-h_0} = \sigma(\mathcal{X}_{t-1}, \cdots, \mathcal{X}_{t-h_0})$ with a prespecified positive integer h_0 which is commonly used in the literature; see Lam et al. (2011), Wang et al. (2019), Chen et al. (2021) among others. Next, we shall suggest a new metric, the CTMDD, that summarizes the mean dependence information between $\mathcal{X}_t$ and $\mathcal{F}_{t-1,t-h_0}$ in a pairwise fashion.

Definition 3.3. For $\mathbb{E}(\|\mathcal{X}_t\|_F^2 + |K(\mathcal{X}_t, \mathbf{0})|^2) < \infty$, the mode-$k$ cumulative tensor martingale difference divergence matrix (CTMDDM), $\mathbf{M}_{h_0}^{(k)}$, is defined as

$$\begin{aligned}\mathbf{M}_{h_0}^{(k)} &= \sum_{h=1}^{h_0} \left\{-\mathbb{E}\left[\{(\mathcal{X}_t)_{(k)} - \mu_{(k)}\}\{(\mathcal{X}'_t)_{(k)} - \mu_{(k)}\}^\top K(\mathcal{X}_{t-h}, \mathcal{X}'_{t-h})\right]\right\} \\ &= \sum_{h=1}^{h_0} \Psi_h^{(k)}, \end{aligned} \tag{7}$$

where $\Psi_h^{(k)} = -\mathbb{E}\left[\{(\mathcal{X}_t)_{(k)} - \mu_{(k)}\}\{(\mathcal{X}'_t)_{(k)} - \mu_{(k)}\}^\top K(\mathcal{X}_{t-h}, \mathcal{X}'_{t-h})\right]$, $(\mathcal{X}'_t, \mathcal{X}'_{t-h})$ is an independent copy of $(\mathcal{X}_t, \mathcal{X}_{t-h})$, $(\mathcal{X}_t)_{(k)} \in \mathbb{R}^{r_k \times \Pi_{j \neq k} r_j}$ is the mode-k matricization of $\mathcal{X}_t$, and $\mu_{(k)} = \mathbb{E}\{(\mathcal{X}_t)_{(k)}\}$, and $K(\cdot, \cdot)$ is a distance of strong negative type (Lyons, 2013) for a tensor. Collectively, we define the cumulative tensor martingale difference divergence (CTMDD) as the set,

$$\mathcal{M}_{h_0} = \left\{\mathbf{M}_{h_0}^{(1)}, \cdots, \mathbf{M}_{h_0}^{(m)}\right\}.$$

Similar to $\mathbf{\Gamma}_{h_0}$, $\mathbf{M}_{h_0}^{(k)}$ is a $r_k \times r_k$ real, symmetric, and positive semi-definite matrix. It is worth noting that the CTMDD $\mathcal{M}_{h_0}$ is defined for a general distance $K(\cdot, \cdot)$, where a metric space $(\mathcal{X}_t; K)$ has strong negative type; see Lyons (2013) and Chakraborty and Zhang (2019) for more discussion on the strong negative type. One natural example of $K(\cdot, \cdot)$ is Frobenius norm, i.e., $K_1(\mathcal{X}, \mathcal{X}') = \|\mathcal{X} - \mathcal{X}'\|_F$, $\mathcal{X} \in \mathbb{R}^{r_1 \times \cdots \times r_m}$. To efficiently quantify the mean dependence between two tensors that often have large dimensions, we adopt one distance in Chakraborty and Zhang (2019) and employ $K_2(\mathcal{X}, \mathcal{X}') = \sqrt{\sum_i \|(\mathcal{X}_{(k)})_{\cdot i} - (\mathcal{X}'_{(k)})_{\cdot i}\|}$, where $(\mathcal{X}_{(k)})_{\cdot i}$ is the i-th column of $\mathcal{X}_{(k)}$. On the other hand, we consider another distance which is related to Mahalanobis distance that shows some advantages under our simulation study, i.e., $K_3(\mathcal{X}, \mathcal{X}') = \|\mathcal{Z} - \mathcal{Z}'\|_F$, where $\mathcal{Z} = [\![\mathcal{X}; \mathbf{\Sigma}_1^{-1/2}, \cdots, \mathbf{\Sigma}_m^{-1/2}]\!]$ and $\mathbf{\Sigma}_k = \mathbb{E}\left\{(\mathcal{X}_{(k)} - \tilde{\mu}_k)(\mathcal{X}_{(k)} - \tilde{\mu}_k)^\top\right\}$, $\tilde{\mu}_k = \mathbb{E}(\mathcal{X}_{(k)})$, $k = 1, \cdots, m$. In practice, when the dimension of the data is large, the distance K_2 or K_3 are preferred instead of K_1 based on our simulation study in Section 4. Also, if there is a prior knowledge regarding the group information in the data,

K_2 can easily incorporate such a group information into the distance. For example, when different economic indices for different countries are analyzed, different countries can be grouped based on the continent information. More specifically, if the mode-2 represents the countries and there exists three different continents in the data, let's say $(\mathcal{X}_t)_{(1)} = \left((\mathcal{X}_{1,t})_{(1)}, (\mathcal{X}_{2,t})_{(1)}, (\mathcal{X}_{3,t})_{(1)}\right) \in \mathbb{R}^{r_1 \times r_2}$, $(\mathcal{X}_{i,t})_{(1)} \in \mathbb{R}^{r_1 \times r_{i,2}}$, $\sum_{i=1}^{3} r_{i,2} = r_2$. Then we can use this group information and define

$$K_2(\mathcal{X}_t, \mathcal{X}'_t) = \sqrt{\left\|(\mathcal{X}_{1,t})_{(1)} - (\mathcal{X}'_{1,t})_{(1)}\right\| + \left\|(\mathcal{X}_{2,t})_{(1)} - (\mathcal{X}'_{2,t})_{(1)}\right\| + \left\|(\mathcal{X}_{3,t})_{(1)} - (\mathcal{X}'_{3,t})_{(1)}\right\|}.$$

Furthermore, if the scales of the elements are quite different or show some correlations from the data, the Mahalanobis distance K_3 would be preferred since the Mahalanobis distance measures the fair distance after taking into account the scale differences and the correlation.

The CTMDD is defined by collecting the mean dependence between $\mathcal{X}_t$ and $\mathcal{F}_{t-1,t-h_0}$ in a pairwise fashion. As mentioned in Section 4 of Lee and Shao (2018), it is possible to consider the mean dependence between $\mathcal{X}_t$ and $\mathcal{F}_{t-1,t-h_0}$ jointly and propose a variant of the CTMDD. However, based on our numerical study which is reported in our supplementary material, dimension reduction with a variant matrix shows a very comparable but slightly less accurate performance. Thus, we shall only present the numerical results of the pairwise approach with $\boldsymbol{\mathcal{M}}_{h_0}$ in this main paper; see the supplementary material for more details regarding the joint approach.

Under the condition that $(\mathcal{X}_t; K)$ has strong negative type and by following the arguments in Lyons (2013), we can show that $\mathbf{M}^{(k)}_{h_0}$ indeed maintains the key property of $\boldsymbol{\Gamma}_{h_0}$ and establish the property of $\boldsymbol{\mathcal{M}}_{h_0}$.

Proposition 3.1. *For* $\mathbb{E}(\|\mathcal{X}_t\|_F^2 + |K(\mathcal{X}_t, \mathbf{0})|^2) < \infty$ *and* $k = 1, \cdots, m$,

1. *There exist* $r_k - d_k$ *linearly independent combinations of* $(\mathcal{X}_t)_{(k)}$ *such that they are mean independent of* $\{\mathcal{X}_{t-h}\}_{h=1}^{h_0}$ *if and only if* $\operatorname{rank}\left(\mathbf{M}^{(k)}_{h_0}\right) = d_k$.
2. *Let* $\alpha \in \operatorname{span}^{\perp}\left(\mathbf{M}^{(k)}_{h_0}\right)$, *then* $\mathbb{E}(\mathcal{X}_t \times_{(k)} \alpha^{\top} \mid \mathcal{X}_{t-h}) = \mathbb{E}(\mathcal{X}_t \times_{(k)} \alpha^{\top})$ *a.s. for* $h = 1, \cdots, h_0$.
3. *If* K *is Frobenius norm* K_1, $\operatorname{span}\left(\mathbf{M}^{(k)}_{h_0}\right) = \sum_i \operatorname{span}\left(\boldsymbol{\Gamma}^i_{h_0}\right)$, *where* $\boldsymbol{\Gamma}^i_{h_0} \in \mathbb{R}^{r_k \times r_k}$ *is the CMDDM with* $(\mathcal{X}_t)^{(k)}_{\cdot i}$ *and* $\operatorname{vec}(\mathcal{X}_{t-h})$ *where* $(\mathcal{X}_t)^{(k)}_{\cdot i}$ *is the* i*-th column of* $(\mathcal{X}_t)_{(k)}$.

Remark 3.3. The first and second assertions in Proposition 3.1 generalize Lemma 2.1 of Lee and Shao (2018) in two directions which seem to be nontrivial. One is extending the approach in Lee and Shao (2018) to a tensor time series. The other direction is generalizing the approach of Lee and Shao (2018) by adopting a general class of distances K which is a strong negative type. Proposition 3.1 guarantees that the generalized metric certainly measures the full mean dependence between the current and the past tensor time series up to lag h_0. This can be viewed as an analog of Theorem 3.11 in Lyons (2013), whereas the theoretical argument is noticeably different. Also, this suggests that $\mathbf{M}^{(k)}_{h_0}$ contains the number and the linear forms of the tensor time series $(\mathcal{X}_t)_{(k)}$ that are mean independent of $\mathcal{F}_{t-1,t-h_0}$. We further remark that the subspace $\operatorname{span}\left(\mathbf{M}^{(k)}_{h_0}\right)$ is certainly related to $\operatorname{span}(\mathbf{C}_k)$ and it belongs to $\operatorname{span}(\mathbf{C}_k)$ for $k = 1, \cdots, m$ under the factor model in (2). The third assertion in Proposition 3.1 states the connection between $\mathbf{M}^{(k)}_{h_0}$ and the vector counterpart $\boldsymbol{\Gamma}_{h_0}$. It indicates that the mean dependence information contained in $\boldsymbol{\Gamma}_{h_0}$ is all accumulated in $\boldsymbol{\mathcal{M}}_{h_0}$ when $K = K_1$. It is also worth pointing out another approach to utilize $\boldsymbol{\Gamma}_{h_0}$ to measure the mean dependence for tensor time series. We can compute the CMDDM $\widetilde{\boldsymbol{\Gamma}}_{h_0} \in \mathbb{R}^{\prod_k r_k \times \prod_k r_k}$ with vectorized tensor time series $\{\operatorname{vec}(\mathcal{X}_t)\}_{t=1}^{n}$. This method is related to the vectorized factor model in (2) of Wang et al. (2019). However, this approach cannot preserve the meaningful structure of the tensor time series thus may lose some inter-relationship which appears among each mode and could lead to a loss of interpretation. Furthermore, this approach has more number of parameters for the factor loading matrices compared to our approach under (2); see more discussions on the vectorized factor model in Wang et al. (2019). Lastly, we shall mention about the user-chosen number h_0. Similar to the method of Wang et al. (2019), any h_0 can be selected for our approach if the rank of any $\{\boldsymbol{\Psi}^{(k)}_h\}_{h=1}^{h_0}$ is d_k since this will make the rank of $\mathbf{M}^{(k)}_{h_0}$ be d_k. Selecting h_0 has been a common question in the literature and proposing a method to select h_0 is beyond the scope of this paper. As mentioned in Wang et al. (2019), Lam and Yao (2012), generally, relatively small h_0 is used since major dependence is often at the short time lag and more noises can be added if large h_0 is selected. As suggested in the existing literature, we shall use small h_0 and follow their approach. Considering that most of the existing methods rely on a linear covariance matrix which can only measure the linear dependence, we view our approach can be a useful addition to the dimension reduction method of tensor time series since our proposed method can also handle nonlinear mean dependence.

Inspired by the estimation of $\boldsymbol{\Gamma}_{h_0}$ in Lee and Shao (2018), we construct the estimator of $\mathbf{M}^{(k)}_{h_0}$ by

$$\widehat{\mathbf{M}}^{(k)}_{h_0} = \sum_{h=1}^{h_0} \left\{ \frac{-1}{(n-h)^2} \sum_{t_1, t_2 = h+1}^{n} \{(\mathcal{X}_{t_1})_{(k)} - (\overline{\mathcal{X}})_{(k)}\}\{(\mathcal{X}_{t_2})_{(k)} - (\overline{\mathcal{X}})_{(k)}\}^{\top} K(\mathcal{X}_{t_1-h}, \mathcal{X}_{t_2-h}) \right\},$$

where $(\overline{\mathcal{X}})_{(k)}$ is the sample mean of $(\mathcal{X}_t)_{(k)}$ based on $\{(\mathcal{X}_{h+1})_{(k)}, \cdots, (\mathcal{X}_n)_{(k)}\}$.

3.3. Estimation

We introduce and establish the theoretical results for our estimation procedure using the CTMDD $\mathcal{M}_{h_0}$. We denote $\{\lambda_j^{(k)}, \gamma_j^{(k)}\}$ and $\{\widehat{\lambda}_j^{(k)}, \widehat{\gamma}_j^{(k)}\}$ as eigenvalues in the descending order and the corresponding eigenvectors of $\mathbf{M}_{h_0}^{(k)}$ and $\widehat{\mathbf{M}}_{h_0}^{(k)}$, respectively. We have two specific goals which are seeking the number and the linear transformations of $(\mathcal{X}_t)_{(k)}$ that are mean independent of $\mathcal{F}_{t-1,t-h_0}$. By Proposition 3.1, those information are contained in $\mathbf{M}_{h_0}^{(k)}$. We suggest to estimate $\text{span}^{\perp}(\mathbf{C}_k) = \text{span}(\mathbf{C}_{k,0})$ as the space spanned by the eigenvectors of $\mathbf{M}_{h_0}^{(k)}$ corresponding to zero eigenvalues. As $\mathbf{C}_k$ and $\mathbf{C}_{k,0}$ are only identifiable up to $\text{span}(\mathbf{C}_k)$ and $\text{span}(\mathbf{C}_{k,0})$, respectively, we define the following distance and show the theoretical result.

$$\mathcal{D}(\mathbf{C}_k, \widehat{\mathbf{C}}_k) = \|\mathbf{P}_{\mathbf{C}_k} - \mathbf{P}_{\widehat{\mathbf{C}}_k}\|_F, \tag{8}$$

where $\mathbf{P}_{\mathbf{C}_k}$, $\mathbf{P}_{\widehat{\mathbf{C}}_k}$ are the projection matrices of $\mathbf{C}_k$, $\widehat{\mathbf{C}}_k$, respectively. Notice that $\mathcal{D}(\mathbf{C}_k, \widehat{\mathbf{C}}_k) = 0$ if and only if $\text{span}(\mathbf{C}_k) = \text{span}(\widehat{\mathbf{C}}_k)$. We make the following assumptions, under which we establish the consistency of $\left\{\widehat{\lambda}_j^{(k)}, \widehat{\gamma}_j^{(k)}, \text{span}(\widehat{\mathbf{C}}_k)\right\}$.

Condition 3.1. We assume that $\lambda_1^{(k)} > \lambda_2^{(k)} > \cdots > \lambda_{d_k}^{(k)} > 0 = \lambda_{d_k+1}^{(k)} = \cdots = \lambda_{r_k}^{(k)}$ for $k = 1, \cdots, m$.

Condition 3.2. Let $\{\text{vec}(\mathcal{X}_t)\}$ be a strictly stationary and β-mixing process. We assume that there exists $\delta > 0$ such that $\mathbb{E}\|\mathcal{X}_t\|_F^{6+3\delta} < \infty$ and $\mathbb{E}|K(\mathcal{X}_t, \mathbf{0})|^{6+3\delta} < \infty$. For a $\delta' \in (0, \delta)$, $\beta(n) = O\left(n^{-\frac{2+\delta'}{\delta'}}\right)$.

Condition 3.3. Let $\{\text{vec}(\mathcal{X}_t)\}$ be a strictly stationary and m-dependent process, and $\mathbb{E}\|\mathcal{X}_t\|_F^6 < \infty$, $\mathbb{E}|K(\mathcal{X}_t, \mathbf{0})|^6 < \infty$.

Condition 3.1, Condition 3.2, and Condition 3.3 are analogous to (C1), (C2), and (C2') in Lee and Shao (2018), which are stated for the CTMDD. Condition 3.1 provides us that there is a single eigenvector corresponding to each nonzero eigenvalue which simplifies the derivations. Condition 3.2 and Condition 3.3 are the assumptions imposed on the data $\mathcal{X}_t$ with a general distance $K(\cdot, \cdot)$ which can be easily verified in practice. Those conditions guarantee that the sample estimate $\widehat{\mathbf{M}}_{h_0}^{(k)}$ is close to the $\mathbf{M}_{h_0}^{(k)}$. In other existing methods, the assumptions are imposed on the latent factor $\mathcal{Z}_t$ or the factor loading matrices $\{\mathbf{C}_i\}_{i=1}^m$. However, for our approach, the assumptions are made towards the data $\mathcal{X}_t$ instead, and study the asymptotic properties.

Theorem 3.1. *For $k = 1, \cdots, m$,*

1. *Under the assumptions in Condition 3.1 and Condition 3.2, we have $\left|\widehat{\lambda}_j^{(k)} - \lambda_j^{(k)}\right| = O_p\left(n^{-1/2}\right)$ and $\left\|\widehat{\gamma}_j^{(k)} - \gamma_j^{(k)}\right\| = O_p\left(n^{-1/2}\right)$ for $j = 1, \cdots, d_k$.*
2. *Under the assumptions in Condition 3.1 and Condition 3.3, we have $\widehat{\lambda}_j^{(k)} = O_p\left(n^{-1}\right)$ for $j = d_k + 1, \cdots, r_k$.*
3. *Under the assumptions in Condition 3.1, Condition 3.2, and $(d_1, \cdots, d_m)$ are known, we have*

$$\mathcal{D}(\mathbf{C}_k, \widehat{\mathbf{C}}_k) = O_p(n^{-1/2}).$$

Theorem 3.1 shows that the empirical eigenvalues and eigenvectors of $\widehat{\mathbf{M}}_{h_0}^{(k)}$ are reasonable estimators of their population counterparts for large sample size. Also, when the true dimension d_k is given, we have the consistency of the estimated space $\text{span}(\widehat{\mathbf{C}}_k)$ in terms of the distance $\mathcal{D}$.

Remark 3.4. It is worth mentioning that Theorem 3.1 is developed for fixed $r_1, \cdots, r_m$. It is plausible to extend the consistency results in Theorem 3.1 under the assumptions in Condition 3.2 and Condition 3.3 for moderately growing $r_1, \cdots, r_m$ with n while $d_1, \cdots, d_m$ are fixed, i.e., $r_k^2 n^{-1} \to 0$ as $n \to \infty$ for $k = 1, \cdots, m$. Our proof requires that $\|\widehat{\mathbf{M}}_{h_0}^{(k)} - \mathbf{M}_{h_0}^{(k)}\|^2 \to 0$, where $\|\widehat{\mathbf{M}}_{h_0}^{(k)} - \mathbf{M}_{h_0}^{(k)}\|^2 \le \|\widehat{\mathbf{M}}_{h_0}^{(k)} - \mathbf{M}_{h_0}^{(k)}\|_F^2 \le \sum_{i=1}^{r_k}\sum_{j=1}^{r_k} |(\widehat{\mathbf{M}}_{h_0}^{(k)})_{ij} - (\mathbf{M}_{h_0}^{(k)})_{ij}|^2 = O_p(r_k^2 n^{-1})$, and $(\widehat{\mathbf{M}}_{h_0}^{(k)})_{ij}$, $(\mathbf{M}_{h_0}^{(k)})_{ij}$ are the (i, j)-th entry of $\widehat{\mathbf{M}}_{h_0}^{(k)}$, $\mathbf{M}_{h_0}^{(k)}$. Thus, under the condition that $r_k^2 n^{-1} \to 0$, here $\|\widehat{\mathbf{M}}_{h_0}^{(k)} - \mathbf{M}_{h_0}^{(k)}\|^2 \to 0$ remains valid and allows us to obtain the consistency results. For the rates of convergence, it might be possible to derive explicit rates by considering different scenarios which depend on the strengths of the signal similar to the theoretical results in Wang et al. (2019) and Chen et al. (2021). Then the assumptions shall be made on the latent factors $\mathcal{Z}_t$ and factor loading matrices $\{\mathbf{C}_k\}_{k=1}^m$ instead of the assumptions on the data $\mathcal{X}_t$. In this paper, we shall work under the assumptions made on $\mathcal{X}_t$ in Condition 3.2 and Condition 3.3, and the investigation on the rates of convergence for growing $r_1, \cdots, r_m$ are left for the future study.

In practice, the dimensions $d_1, \cdots, d_m$ are unknown and need to be estimated. To estimate the dimensions, we shall adopt the ratio-based estimator following Wang et al. (2019) and Lam and Yao (2012), i.e., for $k = 1, \cdots, m$,

$$\widehat{d}_k = \text{argmin}_{j=1,\cdots,R} \frac{\widehat{\lambda}_{j+1}^{(k)}}{\widehat{\lambda}_j^{(k)}},$$

where $R = \lfloor r_k/2 \rfloor$.

Based on the results in Theorem 3.1, the ratio-based estimator ensures that $\widehat{d}_k \geq d_k$ in probability. Thus, it will not under select the true dimension. The main reason that we cannot obtain the consistency result for the ratio-based estimator is because we do not know the rate of convergence for $\frac{\widehat{\lambda}_{j+1}^{(k)}}{\widehat{\lambda}_j^{(k)}}$ when $j > d_k$ and cannot guarantee that the minimum of $\frac{\widehat{\lambda}_{j+1}^{(k)}}{\widehat{\lambda}_j^{(k)}}$ is reached at d_k. It is possible to use an alternative method to select the dimensions. For instance, we can apply the ridge-type estimator (Xia et al., 2015), the growth ratio estimator (Ahn and Horenstein, 2013), the transformed contribution ratio estimator (Xia et al., 2017). However, we use the ratio-based estimator following the approach in Wang et al. (2019) and find out that the ratio-based estimator is simple to implement that provides reasonable performance; see Section 4.

4. Simulation

In this section, we study the finite sample performance of our dimension reduction approach with three different distances, K_1, K_2, K_3 in Section 3.2. We compare our approach with the closely related methods, the factor model for a matrix time series in Wang et al. (2019) and the TOPUP approach for a tensor time series in Chen et al. (2021) which shows the most favorable finite sample performances in their simulation study. Notice that the method of Wang et al. (2019) is a special case for the TOPUP approach in Chen et al. (2021). When $m = 2$, the method in Chen et al. (2021) becomes identical to the approach in Wang et al. (2019). Thus, we shall compare our approach with the existing method in Wang et al. (2019) when $m = 2$ and compare our method with the approach in Chen et al. (2021) when $m > 2$. In our simulations, we set the dimension $r_1 = r_2 = 20,\ 50$ or $r_1 = r_2 = r_3 = 20$ and the sample size $n = 20,\ 50,\ 200$. We consider $h_0 = 1,\ 2$ following the simulation study in Chen et al. (2021). For each example, we replicate the simulation 100 times and use the criteria $D(\mathbf{C}_k, \widehat{\mathbf{C}}_k)$ in (8) to measure the accuracy of the dimension reduction method. Notice that the smaller D indicates more accurate dimension reduction result. We first apply the dimension reduction method by setting the dimensions of the factor at the truth and carry out separate simulations to assess the performance of estimating the dimensions using the ratio-based method in Section 3.3.

Example 4.1. The 3-by-2 factor series are generated by the AR(1) models and those are adopted from Wang et al. (2019).

$$\mathcal{Z}_{i,t} = \phi_i \mathcal{Z}_{i,t-1} + \eta_{i,t},\ i = 1, \cdots, 6,$$

where $\phi = (\phi_i)_{i=1}^6 = (-0.5, 0.6, 0.8, -0.4, 0.7, 0.3)^\top$ and $(\eta_{i,t})_{i=1}^6$ are generated from the standard normal distribution. The data is generated by $\mathcal{X}_t = \mathbf{C}_1 \mathcal{Z}_t \mathbf{C}_2^\top + \mathcal{E}_t$, where $\mathcal{Z}_t = \begin{pmatrix} \mathcal{Z}_{1,t} & \mathcal{Z}_{2,t} \\ \mathcal{Z}_{3,t} & \mathcal{Z}_{4,t} \\ \mathcal{Z}_{5,t} & \mathcal{Z}_{6,t} \end{pmatrix} \in \mathbb{R}^{d_1 \times d_2}$, and $\mathbf{C}_1 \in \mathbb{R}^{r_1 \times d_1}$, $\mathbf{C}_2 \in \mathbb{R}^{r_2 \times d_2}$ are generated from $U(-1, 1)$. The error $\text{vec}(\mathcal{E}_t)$ is generated from $N(\mathbf{0}, 0.25\mathbf{\Sigma}_{r_2} \otimes \mathbf{\Sigma}_{r_1})$, where the diagonals of $\mathbf{\Sigma}_{r_1}$, $\mathbf{\Sigma}_{r_2}$ are all 1 and the off-diagonals are 0.2. In this example, since the data is Gaussian and the error $\mathcal{E}_t$ is white noise and martingale difference, two approaches shall estimate the identical factor loading spaces, thus the dimensions of factors in model (6) and model (2) are identical, i.e., $d_1 = 3,\ d_2 = 2$ and $v_1 = 3,\ v_2 = 2$.

Table 1 summarizes the averages and standard deviations of $D(\mathbf{C}_1, \widehat{\mathbf{C}_1})$ and $D(\mathbf{C}_2, \widehat{\mathbf{C}_2})$. Both approaches produce smaller D-distance which implies that all methods precisely estimate the targeted subspaces. We notice that the data is generated by Gaussian linear time series model where all the dependence can be captured by the linear metric. Thus, the existing method is expected to perform well. However, it is interesting to observe that our approach outperforms the existing method in terms of the smaller D-distance when the sample size is small, i.e., $n = 20,\ 50$. It appears that when $n = 200$, our approach and the existing method are comparable with our approach slightly performing better than the existing method. Among different distances for our approach, our methods with K_2 and K_3 tend to produce smaller D-distance than the one with K_1. Overall, we observe that D-distance decreases as n increases or the dimensions r_1 and r_2 increase, where the latter phenomenon is often called the blessing of dimensionality. Also, the performances for all methods do not change substantially with different values of h_0, which shows that both methods are less sensitive to the choice of h_0. Table 2 shows that the ratio-based estimator is correctly identifying the true dimension of the factor most of the time.

Example 4.2. In this example, we consider a matrix time series with a factor which is white noise but not martingale difference. Thus, the error series in (6) and (2) are different. The 3-by-2 factor series are generated by the all-pass ARMA(1,1) models.

$$\mathcal{Z}_{i,t} = \phi_i \mathcal{Z}_{i,t-1} - \phi_i^{-1} \eta_{i,t-1} + \eta_{i,t},\ i = 1, \cdots, 6,$$

where $\phi = (\phi_i)_{i=1}^6 = (0.3, 0.6, 0.1, 0.4, 0.7, 0.5)^\top$ and $(\eta_{i,t})_{i=1}^6$ are generated by $t(10)$. The data is generated by $\mathcal{X}_t = \mathbf{C}_1 \mathcal{Z}_t \mathbf{C}_2^\top + \mathcal{E}_t$, with $\mathbf{C}_1$ and $\mathbf{C}_2$ defined in Example 4.1. The error $\text{vec}(\mathcal{E}_t)$ is generated from $N(\mathbf{0}, 0.25 I_{r_2} \otimes I_{r_1})$. Note that the factors $\mathcal{Z}_t$ are white noise

Table 1
Average and standard deviation of $D(\mathbf{C}_1, \widehat{\mathbf{C}}_1)$ and $D(\mathbf{C}_2, \widehat{\mathbf{C}}_2)$. Two methods are compared: the factor model in Wang et al. (2019) ($\mathcal{L}_{h_0}$) and our approach ($\mathcal{M}_{h_0}$).

			$h_0 = 1$			
			$\mathcal{L}_{h_0}$	$\mathcal{M}_{h_0}$		
				K_1	K_2	K_3
$D(\mathbf{C}_1, \widehat{\mathbf{C}}_1)$	$n = 20$	$r_1 = 20,\ r_2 = 20$	0.1447 (0.0632)	0.1014 (0.0264)	0.0928 (0.0244)	0.0937 (0.0260)
		$r_1 = 50,\ r_2 = 50$	0.0893 (0.0300)	0.0610 (0.0117)	0.0530 (0.0101)	0.0507 (0.0103)
	$n = 50$	$r_1 = 20,\ r_2 = 20$	0.0749 (0.0173)	0.0649 (0.0168)	0.0605 (0.0161)	0.0665 (0.0165)
		$r_1 = 50,\ r_2 = 50$	0.0505 (0.0128)	0.0394 (0.0077)	0.0340 (0.0067)	0.0325 (0.0076)
	$n = 200$	$r_1 = 20,\ r_2 = 20$	0.0331 (0.0058)	0.0315 (0.0052)	0.0303 (0.0052)	0.0491 (0.0078)
		$r_1 = 50,\ r_2 = 50$	0.0214 (0.0027)	0.0203 (0.0030)	0.0190 (0.0028)	0.0213 (0.0034)
$D(\mathbf{C}_2, \widehat{\mathbf{C}}_2)$	$n = 20$	$r_1 = 20,\ r_2 = 20$	0.0846 (0.0227)	0.0649 (0.0133)	0.0548 (0.0107)	0.0504 (0.0101)
		$r_1 = 50,\ r_2 = 50$	0.0461 (0.0117)	0.0362 (0.0073)	0.0322 (0.0071)	0.0311 (0.0077)
	$n = 50$	$r_1 = 20,\ r_2 = 20$	0.0586 (0.0189)	0.0451 (0.0111)	0.0366 (0.0076)	0.0314 (0.0069)
		$r_1 = 50,\ r_2 = 50$	0.0312 (0.0076)	0.0248 (0.0054)	0.0217 (0.0047)	0.0213 (0.0047)
	$n = 200$	$r_1 = 20,\ r_2 = 20$	0.0298 (0.0072)	0.0257 (0.0056)	0.0223 (0.0046)	0.0196 (0.0047)
		$r_1 = 50,\ r_2 = 50$	0.0151 (0.0028)	0.0134 (0.0022)	0.0123 (0.0023)	0.0161 (0.0026)
			$h_0 = 2$			
			$\mathcal{L}_{h_0}$	$\mathcal{M}_{h_0}$		
				K_1	K_2	K_3
$D(\mathbf{C}_1, \widehat{\mathbf{C}}_1)$	$n = 20$	$r_1 = 20,\ r_2 = 20$	0.1330 (0.0551)	0.1003 (0.0275)	0.0939 (0.0257)	0.0944 (0.0267)
		$r_1 = 50,\ r_2 = 50$	0.0782 (0.0210)	0.0582 (0.0113)	0.0527 (0.0103)	0.0513 (0.0102)
	$n = 50$	$r_1 = 20,\ r_2 = 20$	0.0770 (0.0191)	0.0659 (0.0162)	0.0617 (0.0159)	0.0665 (0.0163)
		$r_1 = 50,\ r_2 = 50$	0.0491 (0.0119)	0.0386 (0.0081)	0.0339 (0.0072)	0.0326 (0.0076)
	$n = 200$	$r_1 = 20,\ r_2 = 20$	0.0356 (0.0068)	0.0325 (0.0058)	0.0315 (0.0061)	0.0488 (0.0077)
		$r_1 = 50,\ r_2 = 50$	0.0227 (0.0035)	0.0207 (0.0031)	0.0192 (0.0029)	0.0214 (0.0035)
$D(\mathbf{C}_2, \widehat{\mathbf{C}}_2)$	$n = 20$	$r_1 = 20,\ r_2 = 20$	0.0752 (0.0168)	0.0613 (0.0121)	0.0536 (0.0104)	0.0507 (0.0105)
		$r_1 = 50,\ r_2 = 50$	0.0436 (0.0107)	0.0320 (0.0072)	0.0319 (0.0072)	0.0313 (0.0077)
	$n = 50$	$r_1 = 20,\ r_2 = 20$	0.0566 (0.0147)	0.0424 (0.0096)	0.0349 (0.0071)	0.0315 (0.0070)
		$r_1 = 50,\ r_2 = 50$	0.0311 (0.0081)	0.0249 (0.0060)	0.0219 (0.0050)	0.0213 (0.0047)
	$n = 200$	$r_1 = 20,\ r_2 = 20$	0.0315 (0.0083)	0.0256 (0.0056)	0.0216 (0.0044)	0.0192 (0.0044)
		$r_1 = 50,\ r_2 = 50$	0.0158 (0.0030)	0.0138 (0.0024)	0.0128 (0.0024)	0.0161 (0.0026)

Table 2
Relative frequency of correctly estimating dimension for the factor series. Two methods are compared: the factor model in Wang et al. (2019) ($\mathcal{L}_{h_0}$) and our approach ($\mathcal{M}_{h_0}$).

			$h_0 = 1$				$h_0 = 2$			
			$\mathcal{L}_{h_0}$	$\mathcal{M}_{h_0}$			$\mathcal{L}_{h_0}$	$\mathcal{M}_{h_0}$		
				K_1	K_2	K_3		K_1	K_2	K_3
$\mathbf{C}_1$	$n = 20$	$r_1 = 20,\ r_2 = 20$	0.72	0.92	0.96	1.00	0.81	0.93	0.97	1.00
		$r_1 = 50,\ r_2 = 50$	0.81	0.96	1.00	0.98	0.83	0.99	0.99	0.98
	$n = 50$	$r_1 = 20,\ r_2 = 20$	0.89	0.95	0.99	1.00	0.86	0.94	0.98	1.00
		$r_1 = 50,\ r_2 = 50$	0.97	0.99	0.99	1.00	0.95	0.99	0.99	1.00
	$n = 200$	$r_1 = 20,\ r_2 = 20$	1.00	1.00	1.00	1.00	0.99	1.00	1.00	1.00
		$r_1 = 50,\ r_2 = 50$	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00
$\mathbf{C}_2$	$n = 20$	$r_1 = 20,\ r_5 = 20$	0.87	0.96	0.98	1.00	0.91	0.98	0.99	1.00
		$r_1 = 50,\ r_5 = 50$	0.98	1.00	1.00	1.00	0.98	1.00	1.00	1.00
	$n = 50$	$r_1 = 20,\ r_2 = 20$	0.86	0.94	0.99	1.00	0.89	0.94	0.99	1.00
		$r_1 = 50,\ r_2 = 50$	0.97	0.99	1.00	1.00	0.96	0.99	1.00	1.00
	$n = 200$	$r_1 = 20,\ r_2 = 20$	0.98	0.98	1.00	1.00	0.94	0.96	1.00	1.00
		$r_1 = 50,\ r_2 = 50$	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00

but not martingale difference. Hence, the existing method treats $\mathbf{C}_1 \mathcal{Z}_t \mathbf{C}_2^\top$ as part of $\mathcal{W}_t$ in (6) and cannot detect them. Therefore, our approach has a larger number of factors compared to the existing methods. Under our model in (2), our approach has $d_1 = 3,\ d_2 = 2$.

In Table 3 and Table 4, we observe that our approach is superior to the existing approach in terms of having smaller D-distances and higher proportions of correctly identifying the true dimensions in all cases. In this example, we consider factors which are

Table 3
Average and standard deviation of $D(\mathbf{C}_1, \widehat{\mathbf{C}}_1)$ and $D(\mathbf{C}_2, \widehat{\mathbf{C}}_2)$. Two methods are compared: the factor model in Wang et al. (2019) ($\mathcal{L}_{h_0}$) and our approach ($\mathcal{M}_{h_0}$).

			$h_0 = 1$			
			$\mathcal{L}_{h_0}$	$\mathcal{M}_{h_0}$		
				K_1	K_2	K_3
$D(\mathbf{C}_1, \widehat{\mathbf{C}}_1)$	$n = 20$	$r_1 = 20,\ r_2 = 20$	0.3212 (0.1685)	0.1097 (0.0241)	0.0790 (0.0146)	0.0651 (0.0106)
		$r_1 = 50,\ r_2 = 50$	0.1122 (0.0375)	0.0542 (0.0105)	0.0398 (0.0063)	0.0336 (0.0048)
	$n = 50$	$r_1 = 20,\ r_2 = 20$	0.2745 (0.1013)	0.0902 (0.0176)	0.0550 (0.0099)	0.0377 (0.0060)
		$r_1 = 50,\ r_2 = 50$	0.1061 (0.0259)	0.0449 (0.0066)	0.0283 (0.0036)	0.0197 (0.0022)
	$n = 200$	$r_1 = 20,\ r_2 = 20$	0.2542 (0.1019)	0.0840 (0.0181)	0.0444 (0.0096)	0.0193 (0.0023)
		$r_1 = 50,\ r_2 = 50$	0.1100 (0.0257)	0.0402 (0.0071)	0.0220 (0.0036)	0.0096 (0.0007)
$D(\mathbf{C}_2, \widehat{\mathbf{C}}_2)$	$n = 20$	$r_1 = 20,\ r_2 = 20$	0.1537 (0.0602)	0.0626 (0.0157)	0.0436 (0.0090)	0.0356 (0.0064)
		$r_1 = 50,\ r_2 = 50$	0.0708 (0.0195)	0.0352 (0.0063)	0.0249 (0.0040)	0.0202 (0.0030)
	$n = 50$	$r_1 = 20,\ r_2 = 20$	0.1525 (0.0566)	0.0554 (0.0124)	0.0331 (0.0069)	0.0218 (0.0037)
		$r_1 = 50,\ r_2 = 50$	0.0731 (0.0196)	0.0302 (0.0057)	0.0185 (0.0030)	0.0125 (0.0014)
	$n = 200$	$r_1 = 20,\ r_2 = 20$	0.1373 (0.0397)	0.0495 (0.0102)	0.0264 (0.0057)	0.0112 (0.0017)
		$r_1 = 50,\ r_2 = 50$	0.0708 (0.0161)	0.0274 (0.0052)	0.0148 (0.0029)	0.0061 (0.0006)
			$h_0 = 2$			
			$\mathcal{L}_{h_0}$	$\mathcal{M}_{h_0}$		
				K_1	K_2	K_3
$D(\mathbf{C}_1, \widehat{\mathbf{C}}_1)$	$n = 20$	$r_1 = 20,\ r_2 = 20$	0.1961 (0.0663)	0.0946 (0.0189)	0.0737 (0.0129)	0.0652 (0.0108)
		$r_1 = 50,\ r_2 = 50$	0.0858 (0.0203)	0.0471 (0.0083)	0.0376 (0.0058)	0.0339 (0.0048)
	$n = 50$	$r_1 = 20,\ r_2 = 20$	0.1786 (0.0497)	0.0739 (0.0121)	0.0480 (0.0076)	0.0374 (0.0058)
		$r_1 = 50,\ r_2 = 50$	0.0791 (0.0146)	0.0362 (0.0045)	0.0247 (0.0029)	0.0198 (0.0022)
	$n = 200$	$r_1 = 20,\ r_2 = 20$	0.1701 (0.0398)	0.0648 (0.0127)	0.0351 (0.0060)	0.0186 (0.0023)
		$r_1 = 50,\ r_2 = 50$	0.0814 (0.0148)	0.0318 (0.0042)	0.0177 (0.0022)	0.0095 (0.0007)
$D(\mathbf{C}_2, \widehat{\mathbf{C}}_2)$	$n = 20$	$r_1 = 20,\ r_2 = 20$	0.1028 (0.0285)	0.0396 (0.0076)	0.0396 (0.0076)	0.0357 (0.0065)
		$r_1 = 50,\ r_2 = 50$	0.0530 (0.0138)	0.0227 (0.0034)	0.0227 (0.0034)	0.0203 (0.0031)
	$n = 50$	$r_1 = 20,\ r_2 = 20$	0.1013 (0.0286)	0.0280 (0.0052)	0.0281 (0.0052)	0.0216 (0.0037)
		$r_1 = 50,\ r_2 = 50$	0.0501 (0.0098)	0.0159 (0.0021)	0.0159 (0.0021)	0.0125 (0.0014)
	$n = 200$	$r_1 = 20,\ r_2 = 20$	0.0978 (0.0237)	0.0209 (0.0037)	0.0211 (0.0037)	0.0108 (0.0017)
		$r_1 = 50,\ r_2 = 50$	0.0508 (0.0099)	0.0116 (0.0017)	0.0117 (0.0017)	0.0061 (0.0006)

white noise but not martingale difference. Therefore, the existing method with a covariance matrix cannot detect the factors. We speculate that this could be a part of the reason for the larger D-distances produced by the existing approach. This example shows the advantage of our approach when there exists nonlinear dependence which is not detectable by a linear metric.

Example 4.3. We consider a tensor time series with factors in Example 4.2, but 2-by-2-by-2 factor series with $\phi = (\phi_i)_{i=1}^{8} = (0.7, 0.5, 0.2, 0.8, 0.3, 0.6, 0.1, 0.4)^\top$. Similarly, the data is generated by $\mathcal{X}_t = [\![\mathcal{Z}_t; \mathbf{C}_1, \mathbf{C}_2, \mathbf{C}_3]\!] + \mathcal{E}_t$, where $\mathcal{Z}_t \in \mathbb{R}^{d_1 \times d_2 \times d_3}$, and $\mathbf{C}_1 \in \mathbb{R}^{r_1 \times d_1}$, $\mathbf{C}_2 \in \mathbb{R}^{r_2 \times d_2}$, $\mathbf{C}_3 \in \mathbb{R}^{r_3 \times d_3}$ are generated from $U(-1, 1)$. The error $\mathrm{vec}(\mathcal{E}_t)$ is generated from $N(\mathbf{0}, 0.25\mathbf{\Sigma}_{r_3} \otimes \mathbf{\Sigma}_{r_2} \otimes \mathbf{\Sigma}_{r_1})$, where the diagonals of $\mathbf{\Sigma}_{r_1}$, $\mathbf{\Sigma}_{r_2}$, $\mathbf{\Sigma}_{r_3}$ are all 1 and the off-diagonals are 0.4, 0.3, 0.2, respectively. Similar to Example 4.2, we have $d_1 = d_2 = d_3 = 2$ under our factor model.

From Table 5, it is shown that our approach produces smaller D-distances than the existing method which indicates that our approach generates more accurate dimension reduction results. Furthermore, we observe that the performance of the existing method improves greatly for the larger h_0 showing some sensitivity to the choice of h_0 while our approach shows comparable results across different values of h_0. In terms of the dimension selection results, we see that the ratio-based estimator produces reasonable results for our method which are reported in Table 6.

5. Real data illustrations

In this section, we demonstrate the usefulness of our approach in the context of prediction by considering two real data sets which are sales data and NYC taxi data. To compare the prediction performance, we compute the forecasting error after dividing the centered data into training and testing sets. In particular, we measure the accuracy of the h-step ahead prediction by

$$\mathrm{FE} = \frac{1}{r_1 \cdots r_m} \|\mathcal{X}_{i+h} - \widehat{\mathcal{X}}_{i+h}\|_F^2,$$

Table 4
Relative frequency of correctly estimating dimension for the factor series. Two methods are compared: the factor model in Wang et al. (2019) ($\mathcal{L}_{h_0}$) and our approach ($\mathcal{M}_{h_0}$).

			$h_0=1$				$h_0=2$			
			$\mathcal{L}_{h_0}$	$\mathcal{M}_{h_0}$			$\mathcal{L}_{h_0}$	$\mathcal{M}_{h_0}$		
				K_1	K_2	K_3		K_1	K_2	K_3
$\mathbf{C}_1$	$n=20$	$r_1=20,\ r_2=20$	0.18	0.67	0.82	0.86	0.37	0.74	0.84	0.87
		$r_1=50,\ r_2=50$	0.63	0.98	1.00	1.00	0.86	1.00	1.00	1.00
	$n=50$	$r_1=20,\ r_2=20$	0.16	0.61	0.93	0.99	0.35	0.86	0.98	1.00
		$r_1=50,\ r_2=50$	0.64	1.00	1.00	1.00	0.87	1.00	1.00	1.00
	$n=200$	$r_1=20,\ r_2=20$	0.21	0.74	0.94	1.00	0.41	0.84	0.98	1.00
		$r_1=50,\ r_2=50$	0.67	1.00	1.00	1.00	0.88	1.00	1.00	1.00
$\mathbf{C}_2$	$n=20$	$r_1=20,\ r_5=20$	0.56	0.95	0.95	1.00	0.79	0.98	0.98	1.00
		$r_1=50,\ r_5=50$	0.86	1.00	1.00	1.00	0.96	1.00	1.00	1.00
	$n=50$	$r_1=20,\ r_2=20$	0.47	1.00	1.00	1.00	0.71	1.00	1.00	1.00
		$r_1=50,\ r_2=50$	0.85	1.00	1.00	1.00	0.97	1.00	1.00	1.00
	$n=200$	$r_1=20,\ r_2=20$	0.56	1.00	1.00	1.00	0.71	1.00	1.00	1.00
		$r_1=50,\ r_2=50$	0.87	1.00	1.00	1.00	0.96	1.00	1.00	1.00

Table 5
Average and standard deviation of $D(\mathbf{C}_1,\widehat{\mathbf{C}}_1)$, $D(\mathbf{C}_2,\widehat{\mathbf{C}}_2)$, and $D(\mathbf{C}_3,\widehat{\mathbf{C}}_3)$. Two methods are compared: the TOPUP method in Chen et al. (2021) ($\mathcal{L}_{h_0}$) and our approach ($\mathcal{M}_{h_0}$).

		$h_0=1$			
		$\mathcal{L}_{h_0}$	$\mathcal{M}_{h_0}$		
			K_1	K_2	K_3
$D(\mathbf{C}_1,\widehat{\mathbf{C}}_1)$	$n=20$	0.0625 (0.0341)	0.0487 (0.0162)	0.0444 (0.0116)	0.0429 (0.0103)
	$n=50$	0.0641 (0.0331)	0.0454 (0.0134)	0.0423 (0.0097)	0.0414 (0.0083)
	$n=200$	0.0650 (0.0399)	0.0419 (0.0121)	0.0396 (0.0062)	0.0395 (0.0036)
$D(\mathbf{C}_2,\widehat{\mathbf{C}}_2)$	$n=20$	0.0736 (0.0568)	0.0487 (0.0162)	0.0434 (0.0157)	0.0420 (0.0147)
	$n=50$	0.0731 (0.0462)	0.0450 (0.0157)	0.0403 (0.0094)	0.0389 (0.0076)
	$n=200$	0.0671 (0.0404)	0.0447 (0.0120)	0.0395 (0.0063)	0.0374 (0.0038)
$D(\mathbf{C}_3,\widehat{\mathbf{C}}_3)$	$n=20$	0.0339 (0.0139)	0.0186 (0.0055)	0.0144 (0.0039)	0.0131 (0.0035)
	$n=50$	0.0332 (0.0119)	0.0147 (0.0036)	0.0093 (0.0021)	0.0076 (0.0018)
	$n=200$	0.0311 (0.0125)	0.0141 (0.0045)	0.0075 (0.0019)	0.0045 (0.0011)
		$h_0=2$			
		$\mathcal{L}_{h_0}$	$\mathcal{M}_{h_0}$		
			K_1	K_2	K_3
$D(\mathbf{C}_1,\widehat{\mathbf{C}}_1)$	$n=20$	0.0531 (0.0208)	0.0468 (0.0140)	0.0441 (0.0111)	0.0431 (0.0105)
	$n=50$	0.0538 (0.0209)	0.0449 (0.0120)	0.0425 (0.0093)	0.0415 (0.0083)
	$n=200$	0.0534 (0.0243)	0.0406 (0.0081)	0.0395 (0.0048)	0.0395 (0.0036)
$D(\mathbf{C}_2,\widehat{\mathbf{C}}_2)$	$n=20$	0.0574 (0.0332)	0.0458 (0.0175)	0.0428 (0.0151)	0.0418 (0.0143)
	$n=50$	0.0531 (0.0217)	0.0419 (0.0113)	0.0396 (0.0084)	0.0390 (0.0076)
	$n=200$	0.0520 (0.0291)	0.0419 (0.0091)	0.0388 (0.0052)	0.0374 (0.0038)
$D(\mathbf{C}_3,\widehat{\mathbf{C}}_3)$	$n=20$	0.0248 (0.0074)	0.0163 (0.0042)	0.0138 (0.0036)	0.0131 (0.0036)
	$n=50$	0.0235 (0.0070)	0.0120 (0.0031)	0.0086 (0.0019)	0.0077 (0.0018)
	$n=200$	0.0223 (0.0052)	0.0105 (0.0025)	0.0062 (0.0014)	0.0045 (0.0011)

where $\widehat{\mathcal{X}}_{i+h}$ is the h-step ahead prediction of $\mathcal{X}_{i+h}$ based on the training set. Therefore, the smaller value of FE indicates more accurate h-step ahead prediction.

5.1. Sales data

In this section, we consider the sales data from a superstore. This data is available at https://www.kaggle.com/yanachshyogoleva/superstore-sales-dataset/data. Since the daily data is too sparse, we first transform the data into weekly data by calculating the total amount of sales for 17 categories and 4 regions, and take one difference in order to make the data stationary. Thus, we have $\mathcal{X}_t=[x_{i_1,i_2,t}]\in\mathbb{R}^{17\times4}$ with the length of time equals to $n=184$, where $x_{i_1,i_2,t}$ is the total amount of sales for each category and region in week t. We divide the centered data into training and testing sets where the last 37 data observations are set as testing set which are approximately 20% of the data. In order to predict $\mathcal{X}_{i+h}$ and generate $\widehat{\mathcal{X}}_{i+h}$, we follow the approaches in Matteson and Tsay (2011) and Lee and Shao (2018). In particular, we use the rolling-window approach. More specifically, we use the following

Table 6
Relative frequency of correctly estimating dimension for the factor series. Two methods are compared: the TOPUP method in Chen et al. (2021) ($\mathcal{L}_{h_0}$) and our approach ($\mathcal{M}_{h_0}$).

		$h_0=1$				$h_0=2$			
		$\mathcal{L}_{h_0}$	$\mathcal{M}_{h_0}$			$\mathcal{L}_{h_0}$	$\mathcal{M}_{h_0}$		
			K_1	K_2	K_3		K_1	K_2	K_3
$\mathbf{C}_1$	$n=20$	0.80	0.86	0.88	0.90	0.84	0.88	0.88	0.91
	$n=50$	0.79	0.93	0.93	0.94	0.83	0.93	0.92	0.93
	$n=200$	0.82	0.96	0.98	1.00	0.90	1.00	1.00	1.00
$\mathbf{C}_2$	$n=20$	0.53	0.67	0.67	0.75	0.59	0.74	0.77	0.75
	$n=50$	0.43	0.67	0.81	0.87	0.59	0.75	0.83	0.85
	$n=200$	0.38	0.66	0.83	0.92	0.54	0.74	0.91	0.92
$\mathbf{C}_3$	$n=20$	0.88	0.96	0.99	1.00	0.95	0.98	0.99	1.00
	$n=50$	0.84	0.99	1.00	1.00	0.90	0.99	1.00	1.00
	$n=200$	0.85	0.99	1.00	1.00	0.93	1.00	1.00	1.00

Table 7
Average of FE (the averages are multiplied 10^{-3}). Two methods are compared: the factor model in Wang et al. (2019) ($\mathcal{L}_{h_0}$) and our approach ($\mathcal{M}_{h_0}$).

		$\mathcal{L}_{h_0}$	$\mathcal{M}_{h_0}$		
			K_1	K_2	K_3
$h_0=1$	1-step ahead	773.26	721.16	724.19	752.24
	2-step ahead	1190.76	977.91	927.06	972.33
	3-step ahead	1232.50	1012.49	1114.75	1107.34
$h_0=2$	1-step ahead	777.62	738.21	755.30	764.71
	2-step ahead	1182.65	1045.26	965.40	939.79
	3-step ahead	1229.59	1085.64	1150.19	1147.88
$h_0=3$	1-step ahead	770.89	751.24	766.75	769.03
	2-step ahead	1189.74	1088.20	971.17	945.95
	3-step ahead	1236.59	1125.51	1147.47	1125.87

procedure. (1) We first estimate the dimension (d_1, d_2) by using the entire data set. Since we have a moderate dimension for $\mathcal{X}_t$, we apply the ratio-based estimator in Section 3.3 with $R = r_k - 1$. For $h_0 = 1,\ 2,\ 3$, $\widehat{d}_1 = 1,\ \widehat{d}_2 = 3$ are selected. (2) Based on the estimates of (d_1, d_2), we apply a dimension reduction method to $(\mathcal{X}_{i-n_{train}+1}, \cdots, \mathcal{X}_i)$, $i = n_{train}, \cdots, n_{train} + 36$, n_{train} is the sample size of the training set. We obtain the estimated factor series and factor loading matrices. (3) The optimal vector autoregressive model is fitted to the estimated factors. The order is chosen by the Akaike information criterion (AIC) with the maximum order equals to 10. (4) With the fitted model, we generate the h-step ahead prediction of the estimated factor series and multiply the estimated factor loading matrices in order to generate the h-step ahead prediction of $\mathcal{X}_{i+h}$ and calculate FE.

From Table 7, we observe that our approach noticeably outperforms the existing method in Wang et al. (2019) in terms of smaller FE. This indicates that our approach generates more accurate forecasting for the sales data. Furthermore, Table 8, 9 report the estimated factor loading matrices after applying methods to the entire centered data with $h_0 = 1$. Here, we report the estimated factor loadings for our approach with K_1. However the estimated factor loading matrices with K_2 and K_3 are very similar to the ones with K_1 and those are reported in the supplementary material. For the categories, while copiers load heavily on the loading matrix followed by phones for the existing method, our approach selects copiers to have the highest weight followed by machines. For the regions, both methods agree to give highest weights to east, west, and central for the three factor loadings.

5.2. NYC taxi data

In this section, we consider the NYC taxi data which has been analyzed by Chen et al. (2021). This data is available at https://www1.nyc.gov/site/tlc/about/tlc-trip-record-data.page. We select the recent data that contains the daily trip records in Manhattan starting from January 1, 2017 to December 31, 2018 before COVID 19. Similar to Chen et al. (2021), we calculate the total number of rides moving among the zones within each hour. Since the daily data is too sparse, we transform the data into weekly data by calculating the total number of rides occurred in 10 popular zones and busy hours which are 1-11 hours (PM), and take one difference to make the data stationary. Thus, we have $\mathcal{X}_t = [x_{i_1,i_2,i_3,t}] \in \mathbb{R}^{10\times10\times11}$ with the length of time equals to $n = 103$, where $x_{i_1,i_2,i_3,t}$ is the total number of rides from zone i_1 (the pick-up zone) to zone i_2 (the drop-off zone) and the pick-up time i_3 hour in week t. Similar to Section 5.1, we divide the centered data into training and testing sets where the last 20 observations are set as testing set which are approximately 20% of the data. Next, we follow the same procedure in Section 5.1 with the selected dimensions $\widehat{d}_1 = \widehat{d}_2 = \widehat{d}_3 = 1$ for $h_0 = 1,\ 2,\ 3$ and compare the prediction accuracy.

Table 10 summarizes the prediction performances for our approach and the existing method in Chen et al. (2021). It appears that our $\mathcal{M}_{h_0}$-based approach produces smaller forecasting error on average which suggests that our approach produces more accurate

Table 8
Estimated factor loading matrix $\mathbf{C}_1$ with $h_0 = 1$. Two methods are compared: the factor model in Wang et al. (2019) ($\mathcal{L}_{h_0}$) and our approach ($\mathcal{M}_{h_0}$) with K_1.

$\mathcal{L}_{h_0}$						
Accessories	Appliances	Art	Binders	Bookcases	Chairs	Copiers
0.04	-0.03	0.00	-0.03	-0.02	0.02	-0.99
Envelopes	Fasteners	Furnishings	Labels	Machines	Paper	Phones
-0.01	0.00	0.01	0.00	-0.06	0.02	-0.12
Storage	Supplies	Tables				
0.03	0.02	-0.02				
$\mathcal{M}_{h_0}$						
Accessories	Appliances	Art	Binders	Bookcases	Chairs	Copiers
0.01	0.00	0.00	-0.09	-0.06	-0.07	-0.97
Envelopes	Fasteners	Furnishings	Labels	Machines	Paper	Phones
0.00	0.00	-0.01	0.00	-0.19	0.00	-0.08
Storage	Supplies	Tables				
-0.07	0.00	-0.04				

Table 9
Estimated factor loading matrix $\mathbf{C}_2$ with $h_0 = 1$. Two methods are compared: the factor model in Wang et al. (2019) ($\mathcal{L}_{h_0}$) and our approach ($\mathcal{M}_{h_0}$) with K_1.

$\mathcal{L}_{h_0}$				$\mathcal{M}_{h_0}$			
Central	East	South	West	Central	East	South	West
-0.39	-0.91	0.02	0.12	-0.31	-0.95	0.01	0.00
0.84	-0.40	0.04	-0.35	0.32	-0.11	-0.08	0.94
-0.37	0.04	0.17	-0.91	0.89	-0.29	0.06	-0.33

Table 10
Average of FE. Two methods are compared: the factor model in Chen et al. (2021) ($\mathcal{L}_{h_0}$) and our approach ($\mathcal{M}_{h_0}$).

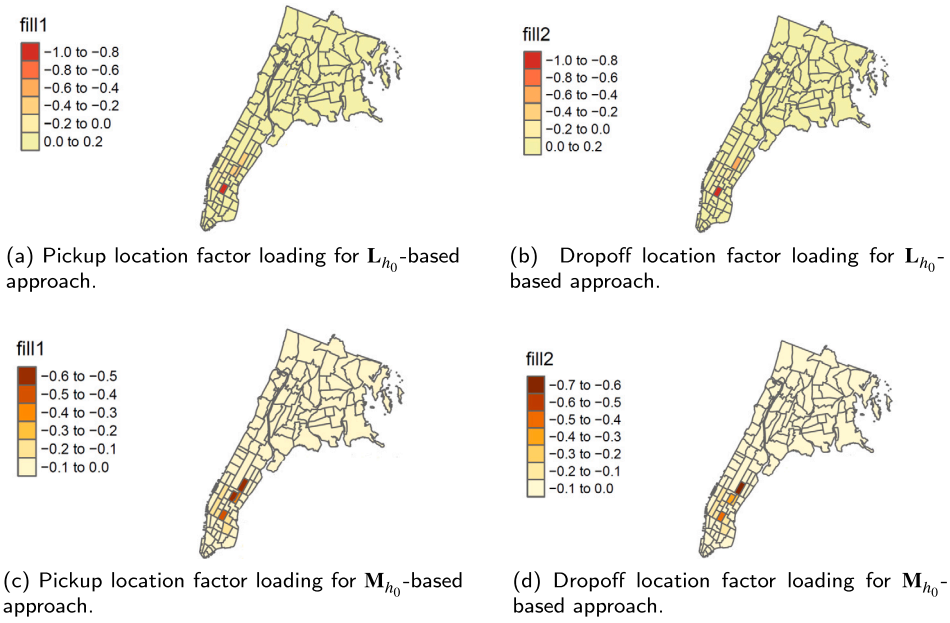
		$\mathcal{L}_{h_0}$	$\mathcal{M}_{h_0}$		
			K_1	K_2	K_3
$h_0 = 1$	1-step ahead	543.08	526.73	526.04	526.28
	2-step ahead	555.94	547.17	546.35	546.55
	3-step ahead	568.06	559.47	558.80	558.91
$h_0 = 2$	1-step ahead	540.45	526.67	525.92	526.09
	2-step ahead	553.89	546.65	546.26	546.32
	3-step ahead	566.58	559.16	558.72	558.81
$h_0 = 3$	1-step ahead	537.80	526.06	525.99	526.31
	2-step ahead	551.67	546.01	546.05	546.34
	3-step ahead	565.36	558.71	558.66	558.88

prediction. It is worth mentioning that the improvement in prediction accuracy by our approach is mainly due to the use of $\mathcal{M}_{h_0}$ since the same modeling procedure is applied to the estimated factor series.

Fig. 1 reports heat maps that summarize the estimated factor loading matrices when $h_0 = 1$ for the existing method and our approach with K_1. The estimated factor loading matrices for our approach with K_2 and K_3 are reported in the supplementary material and they are very similar to Fig. 1c, Fig. 1d. It is interesting to observe that two approaches select different areas as important places. It shows that our $\mathcal{M}_{h_0}$-based approach selects the midtown center, upper east south areas as crucial regions for the pickup locations whereas the existing method chooses union square as the important area for the pickup locations. Also, union square, upper east south are selected as important regions for dropoff locations for our method and the existing method.

6. Conclusion

We propose a new dimension reduction framework for tensor time series by utilizing the CTMDD that can summarize dependence beyond the linear mean dependence. Moreover, we can effectively estimate $d_1, \cdots, d_m$ by employing the ratio-based estimator in Section 3.3. The advantages of our method can be explained by a slightly more general and flexible tensor factor model than the factor model in Wang et al. (2019) and Chen et al. (2021).



(a) Pickup location factor loading for $\mathbf{L}_{h_0}$-based approach.

(b) Dropoff location factor loading for $\mathbf{L}_{h_0}$-based approach.

(c) Pickup location factor loading for $\mathbf{M}_{h_0}$-based approach.

(d) Dropoff location factor loading for $\mathbf{M}_{h_0}$-based approach.

Fig. 1. Heat maps of the estimated factor loading matrices with $h_0 = 1$. Two methods are compared: the factor model in Chen et al. (2021) ($\mathbf{L}_{h_0}$) and our approach ($\mathbf{M}_{h_0}$) with K_1. (For interpretation of the colors in the figure, the reader is referred to the web version of this article.)

For a possible future research direction, it would be interesting to develop an alternative tensor time series dimension reduction based on CP decomposition, following recent advances in Han et al. (2021) and Chang et al. (2021), than factor models.

Acknowledgements

The authors appreciate the editor, the associate editor and the anonymous referees for the constructive comments and suggestions that led to significant improvement of the manuscript. Dr. Zhang's research is supported partly by NSF grant DMS-2053697. Dr. Lee's research is supported by Eugene M. Lang grant.

Appendix A. Supplementary material

Supplementary material related to this article can be found online at https://doi.org/10.1016/j.csda.2024.107998.

References

Ahn, S.C., Horenstein, A.R., 2013. Eigenvalue ratio test for the number of factors. Econometrica 81, 1203–1227.
Chakraborty, S., Zhang, X., 2019. A new framework for distance and kernel-based metrics in high dimensions. Preprint. arXiv:1909.13469.
Chang, J., He, J., Yang, L., Yao, Q., 2021. Modelling matrix time series via a tensor cp-decomposition. Preprint. arXiv:2112.15423.
Chen, E.Y., Fan, J., 2021. Statistical inference for high-dimensional matrix-variate factor models. J. Am. Stat. Assoc., 1–18.
Chen, E.Y., Tsay, R.S., Chen, R., 2020. Constrained factor models for high-dimensional matrix-variate time series. J. Am. Stat. Assoc. 115, 775–793.
Chen, R., Yang, D., Zhang, C.H., 2021. Factor models for high-dimensional tensor time series. J. Am. Stat. Assoc., 1–23.
Cook, R.D., 1998. Regression Graphics: Ideas for Studying Regressions Through Graphics. John Wiley & Sons.
Ding, S., Cook, R.D., 2014. Dimension folding pca and pfc for matrix-valued predictors. Stat. Sin. 24, 463–492.
Gao, Z., Tsay, R.S., 2021. A two-way transformed factor model for matrix-variate time series. Econom. Stat.
Han, Y., Zhang, C.H., Chen, R., 2021. Cp factor model for dynamic tensors. Preprint. arXiv:2110.15517.
Kolda, T.G., Bader, B.W., 2009. Tensor decompositions and applications. SIAM Rev. 51, 455–500.
Lam, C., Yao, Q., 2012. Factor modeling for high-dimensional time series: inference for the number of factors. Ann. Stat., 694–726.
Lam, C., Yao, Q., Bathia, N., 2011. Estimation of latent factors for high-dimensional time series. Biometrika 98, 901–918.
Lee, C.E., Shao, X., 2018. Martingale difference divergence matrix and its application to dimension reduction for stationary multivariate time series. J. Am. Stat. Assoc. 113, 216–229.
Lee, C.E., Shao, X., 2020. Volatility martingale difference divergence matrix and its application to dimension reduction for multivariate volatility. J. Bus. Econ. Stat. 38, 80–92.
Lee, C.E., Zhang, X., Li, L., 2023. Mean dimension reduction and testing for nonparametric tensor response regression. Manuscript.
Li, B., Kim, M.K., Altman, N., et al., 2010. On dimension folding of matrix-or array-valued statistical objects. Ann. Stat. 38, 1094–1121.
Li, L., Zhang, X., 2017. Parsimonious tensor response regression. J. Am. Stat. Assoc. 112, 1131–1146. https://doi.org/10.1080/01621459.2016.1193022.
Lyons, R., 2013. Distance covariance in metric spaces. Ann. Probab. 41, 3284–3305.
Matteson, D.S., Tsay, R.S., 2011. Dynamic orthogonal components for multivariate time series. J. Am. Stat. Assoc. 106, 1450–1463.
Park, J.H., Sriram, T., Yin, X., 2009. Central mean subspace in time series. J. Comput. Graph. Stat. 18, 717–730.
Park, J.H., Sriram, T., Yin, X., 2010. Dimension reduction in time series. Stat. Sin., 747–770.

Rabusseau, G., Kadri, H., 2016. Low-rank regression with tensor responses. In: Proceedings of the 30th International Conference on Neural Information Processing Systems, pp. 1875–1883.
Shao, X., 2011. Testing for white noise under unknown dependence and its applications to diagnostic checking for time series models. Econom. Theory 27, 312–343.
Shao, X., Zhang, J., 2014. Martingale difference correlation and its use in high-dimensional variable screening. J. Am. Stat. Assoc. 109, 1302–1318.
Sheng, W., Yuan, Q., 2020. Sufficient dimension folding in regression via distance covariance for matrix-valued predictors. Stat. Anal. Data Min. ASA Data Sci. J. 13, 71–82.
Sun, W.W., Li, L., 2017. Store: sparse tensor response regression and neuroimaging analysis. J. Mach. Learn. Res. 18, 4908–4944.
Wang, D., Liu, X., Chen, R., 2019. Factor models for matrix-valued high-dimensional time series. J. Econom. 208, 231–248.
Wang, N., Zhang, X., Li, B., 2022. Likelihood-based dimension folding on tensor data. Stat. Sin. 32, 2405–2429.
Xia, Q., Liang, R., Wu, J., 2017. Transformed contribution ratio test for the number of factors in static approximate factor models. Comput. Stat. Data Anal. 112, 235–241.
Xia, Q., Xu, W., Zhu, L., 2015. Consistently determining the number of factors in multivariate volatility modelling. Stat. Sin., 1025–1044.
Zhou, J., Zhu, L., 2021. Modified martingale difference correlations. J. Nonparametr. Stat. 33, 359–386.