CP factor model for dynamic tensors

Yuefeng Han, Dan Yang, Cun-Hui Zhang, Rong Chen

Journal of the Royal Statistical Society Series B: Statistical Methodology, 2024, **86**, 1383–1413
https://doi.org/10.1093/jrsssb/qkae036
Advance access publication 16 May 2024
Original Article

ROYAL STATISTICAL SOCIETY DATA | EVIDENCE | DECISIONS
Journal of the Statistics Society
Series B
Statistical Methodology
B

CP factor model for dynamic tensors

Yuefeng Han[1], Dan Yang[2], Cun-Hui Zhang[3] and Rong Chen[3]

[1]Department of Applied and Computational Mathematics and Statistics, University of Notre Dame, Notre Dame, IN 46556, USA
[2]Faculty of Business and Economics, The University of Hong Kong, Pokfulam, Hong Kong
[3]Department of Statistics, Rutgers University, 110 Frelinghuysen Road, Piscataway, NJ 08854, USA

Address for correspondence: Rong Chen, Department of Statistics, Rutgers University, 110 Frelinghuysen Road, Piscataway, NJ 08854, USA. Email: rongchen@stat.rutgers.edu

Abstract

Observations in various applications are frequently represented as a time series of multidimensional arrays, called tensor time series, preserving the inherent multidimensional structure. In this paper, we present a factor model approach, in a form similar to tensor CANDECOMP/PARAFAC (CP) decomposition, to the analysis of high-dimensional dynamic tensor time series. As the loading vectors are uniquely defined but not necessarily orthogonal, it is significantly different from the existing tensor factor models based on Tucker-type tensor decomposition. The model structure allows for a set of uncorrelated one-dimensional latent dynamic factor processes, making it much more convenient to study the underlying dynamics of the time series. A new high-order projection estimator is proposed for such a factor model, utilizing the special structure and the idea of the higher order orthogonal iteration procedures commonly used in Tucker-type tensor factor model and general tensor CP decomposition procedures. Theoretical investigation provides statistical error bounds for the proposed methods, which shows the significant advantage of utilizing the special model structure. Simulation study is conducted to further demonstrate the finite sample properties of the estimators. Real data application is used to illustrate the model and its interpretations.

Keywords: CANDECOMP/PARAFAC (CP) decomposition, dimension reduction, orthogonal projection, tensor factor model, tensor time series

1 Introduction

In recent years, information technology has made tensors or high-order arrays observations routinely available in applications. For example, such data arises naturally from genomics (Alter & Golub, 2005; Omberg et al., 2007), neuroimaging analysis (Sun & Li, 2017; Zhou et al., 2013), recommender systems (Bi et al., 2018), computer vision (J. Liu et al., 2012), community detection (Anandkumar et al., 2014a), longitudinal data analysis (Hoff, 2015), among others. Most of the developed tensor-based methods were designed for independent and identically distributed (i.i.d.) tensor data or tensor data with i.i.d. noise.

On the other hand, in many applications, the tensors are observed over time, and hence form a tensor-valued time series. For example, the monthly import export volumes of multi-categories of products (e.g. Chemical, Food, Machinery and Electronic, and Footwear and Headwear) among countries naturally form a dynamic sequence of 3-way tensor-variates, each of which representing a weighted directional transportation network. Another example is functional MRI, which typically consists hundreds of thousands of voxels observed over time. A sequence of 2-D or 3-D images can also be modelled as matrix or tensor time series to preserve temporal structure. Development of statistical methods for analysing such large-scale tensor-valued time series is still in its infancy.

Received: October 28, 2021. **Revised:** February 8, 2024. **Accepted:** April 17, 2024

In many settings, although the observed tensors are of high order and high dimension, there is often hidden low-rank structures in the tensors that can be exploited to facilitate the data analysis. Such a low-rank condition provides convenient decomposable structures and has been widely used in tensor data analysis. Two common choices of low-rank tensor structures are CANDECOMP/PARAFAC (CP) structure and multilinear/Tucker structure, and each of them has their respective benefits; see the survey in Kolda and Bader (2009).

In dynamic data, the low-rank structures are often realized through factor models, one of the most effective and popular dimension reduction tools. Over the past few decades, there has been a large body of the literature in the statistics and econometrics communities on factor models for vector time series. An incomplete list of the publications includes Chamberlain and Rothschild (1983), Bai and Ng (2002), Stock and Watson (2002), Bai (2003), Fan et al. (2011, 2013, 2016), Forni et al. (2000, 2004, 2005), Pena and Box (1987), Pan and Yao (2008), Lam et al. (2011), and Lam and Yao (2012). Recently, the factor model approach has been developed for analysing high-dimensional dynamic tensor time series (Chang et al., 2023; E. Y. Chen & Fan, 2023; E. Y. Chen et al., 2020, 2024; R. Chen et al., 2022; Y. Han et al., 2020, 2022; Y. Han & Zhang, 2023; D. Wang et al., 2019). These existing works utilize the Tucker low-rank structure in formulating the factor models. Such Tucker-type tensor factor model is also closely related to separable factor analysis in Fosdick and Hoff (2014) under the array Normal distribution of Hoff (2011).

In this paper, we investigate a tensor factor model with a CP type low-rank structure, called TFM-cp. Specifically, let $\mathcal{X}_t$ be an order K tensor of dimensions $d_1 \times d_2 \times \ldots \times d_K$. We assume

$$\mathcal{X}_t = \sum_{i=1}^{r} w_i f_{it} \boldsymbol{a}_{i1} \otimes \boldsymbol{a}_{i2} \otimes \cdots \otimes \boldsymbol{a}_{iK} + \mathcal{E}_t, \quad t = 1, \ldots, T, \tag{1}$$

where $\otimes$ denotes tensor product, $w_i > 0$ represents the signal strength, $\boldsymbol{a}_{ik}$, $i = 1, \ldots, r$, are unit vectors of dimension d_k, with $\|\boldsymbol{a}_{ik}\|_2 = 1$, $\mathcal{E}_t$ is a noise tensor of the same dimension as $\mathcal{X}_t$, and $\{f_{it}, i = 1, \ldots, r\}$ is a set of uncorrelated univariate latent factor processes. That is, the signal part of the observed tensor at time t is a linear combination of r rank-one tensors, $w_i \boldsymbol{a}_{i1} \otimes \boldsymbol{a}_{i2} \otimes \cdots \otimes \boldsymbol{a}_{iK}$. These rank-one tensors are fixed and do not change over time. Here, $\{\boldsymbol{a}_{ik}, 1 \le i \le r, 1 \le k \le K\}$ are called loading vectors and the loading vectors for each mode, $\{\boldsymbol{a}_{ik}, 1 \le i \le r\}$, are not necessarily orthogonal. The dynamics of the tensor time series are driven by the r univariate latent processes f_{it}. By stacking the fibres of the tensor $\mathcal{X}_t$ into a vector, the TFM-cp can be written as a vector factor model, with r factors and a $d \times r$ (where $d = d_1 \ldots d_K$) loading matrix of a special structure induced by the TFM-cp. More detailed discussion of the model is given in Section 2.

A standard approach for dynamic factor model estimation is through the analysis of the covariance or autocovariance of the observed process. The autocovariance of a TFM-cp process in (1) is also a tensor with a low-rank CP structure. Hence, potentially the estimation of (1) can be done with a tensor CP decomposition procedure. However, tensor CP decomposition is well known to be a notoriously challenging problem as it is in general NP hard to compute and the CP rank is not lower semi-continuous (Håstad, 1990; Hillar & Lim, 2013; Kolda & Bader, 2009). There are a number of works on tensor CP decomposition, which is often called tensor principal component analysis (PCA) in the literature, including alternating least squares (ALSs) (Comon et al., 2009), robust tensor power methods with orthogonal components (Anandkumar et al., 2014b), tensor unfolding approaches (Richard & Montanari, 2014; P.-A. Wang & Lu, 2017), rank-one ALSs (Anandkumar et al., 2014c; Sun et al., 2017), and simultaneous matrix diagonalization (Kuleshov et al., 2015). See also Zhou et al. (2013), M. Wang & Song, 2017, Hao et al. (2020), Wang and Li (2020), Auddy and Yuan (2023b), R. Han et al. (2023), among others. Although these methods can be used directly to obtain the low-rank CP components of the autocovariance tensors, they have been designed for general tensors and do not utilize the special structure embedded in the TFM-cp.

In this paper, we develop a new estimation procedure, named as **H**igh-**O**rder **P**rojection **E**stimators (HOPE), for TFM-cp in (1). The procedure includes a warm-start initialization using a newly developed composite principal component analysis (cPCA), and an iterative simultaneous

orthogonalization (ISO) scheme to refine the estimator. The procedure is designed to take the advantage of the special structure of TFM-cp whose autocovariance tensor has a specific CP structure with components close to being orthogonal and of a high-order coherence in a multiplicative form. The proposed cPCA takes advantage of this feature so the initialization is better than using random projection initialization often used in generic CP decomposition algorithms. The refinement step makes use of the multiplicative coherence again and is better than the ALSs, the iterative projection algorithm (Y. Han et al., 2020), and other forms of the high-order orthogonal iteration (HOOI) (De Lathauwer et al., 2000; Y. Liu et al., 2014; Zhang & Xia, 2018). Our theoretical analysis provides details of these improvements.

In the theoretical analysis, we establish statistical upper bounds on the estimation errors of the factor loading vectors for the proposed algorithms. The cPCA yields useful and good initial estimators with less restrictive conditions, and the iterative algorithm provides faster statistical error rates under weaker conditions than the generic CP decomposition algorithms. For cPCA, the number of factors r can increase with the dimensions of the tensor time series and is allowed to be larger than $\max_k d_k$. We also derive the statistical guarantees of the iterative algorithm under the settings where the tensor is (sufficiently) undercomplete ($r \ll \min_k d_k$). It is worth noting that the iterative refinement algorithm has much sharper upper bounds for the statistical error than the cPCA initial estimators.

The TFM-cp in (1) can also be written as a tensor factor model with a Tucker form (TFM-tucker) of a special structure. See (3) for the definition of TFM-tucker and Remark 3 for the comparison between TFM-cp and TFM-tucker from the perspectives of modelling assumptions and interpretations. Potentially, the iterative estimation procedures designed for TFM-tucker can also be used here Y. Han et al. (2020), ignoring the special TFM-cp structure. However, HOPE has lower computational complexity per iteration, requires less restrictive conditions and exhibits faster convergence rate, by fully utilizing the structure of TFM-cp. See Remark 16 for further discussion. They also share the nice properties that the increase in either the dimensions $d_1, \ldots, d_k$, or the sample size can improve the estimation of the factor loading vectors or spaces.

The rest of the paper is organized as follows. After a brief introduction of the basic notations and preliminaries of tensor analysis in Section 1.1, we introduce a tensor factor model with CP low-rank structure in Section 2. The estimation procedures of the factors and the loading vectors are presented in Section 3. Section 4 investigates the theoretical properties of the proposed methods. Section 5 develops some alternative algorithms to tensor factor models, which extend existing popular CP methods to the autocovariance tensors with cPCA as initialization, and provides some simulation studies to demonstrate the numerical performance of all the estimation procedures. Section 6 illustrates the model and its interpretations in real data applications. Section 7 provides a short concluding remark. All technical details and more simulation results are relegated to the online supplementary materials.

1.1 Notations and preliminaries

The following basic notations and preliminaries will be used throughout the paper. Define $\|x\|_q = (x_1^q + \cdots + x_p^q)^{1/q}$, $q \geq 1$, for any vector $x = (x_1, \ldots, x_p)^\top$. The matrix spectral norm is denoted as

$$\|A\|_S = \max_{\|x\|_2=1, \|y\|_2=1} \|x^\top A y\|_2.$$

For two sequences of real numbers $\{a_n\}$ and $\{b_n\}$, write $a_n = O(b_n)$ (resp. $a_n \asymp b_n$) if there exists a constant C such that $|a_n| \leq C|b_n|$ (resp. $1/C \leq a_n/b_n \leq C$) holds for all sufficiently large n, and write $a_n = o(b_n)$ if $\lim_{n\to\infty} a_n/b_n = 0$. Write $a_n \lesssim b_n$ (resp. $a_n \gtrsim b_n$) if there exists a constant C such that $a_n \leq Cb_n$ (resp. $a_n \geq Cb_n$).

For any two tensors $\mathcal{A} \in \mathbb{R}^{m_1 \times m_2 \times \cdots \times m_K}$, $\mathcal{B} \in \mathbb{R}^{r_1 \times r_2 \times \cdots \times r_N}$, denote the tensor product $\otimes$ as $\mathcal{A} \otimes \mathcal{B} \in \mathbb{R}^{m_1 \times \cdots \times m_K \times r_1 \times \cdots \times r_N}$, such that

$$(\mathcal{A} \otimes \mathcal{B})_{i_1,\ldots,i_K,j_1,\ldots,j_N} = (\mathcal{A})_{i_1,\ldots,i_K} (\mathcal{B})_{j_1,\ldots,j_N}.$$

The k-mode product of $\mathcal{A} \in \mathbb{R}^{r_1 \times r_2 \times \cdots \times r_K}$ with a matrix $U \in \mathbb{R}^{m_k \times r_k}$ is an order K-tensor of size $r_1 \times \cdots \times r_{k-1} \times m_k \times r_{k+1} \times \cdots \times r_K$ and will be denoted as $\mathcal{A} \times_k U$, such that

$$(\mathcal{A} \times_k U)_{i_1,\ldots,i_{k-1},j,i_{k+1},\ldots,i_K} = \sum_{i_k=1}^{r_k} \mathcal{A}_{i_1,i_2,\ldots,i_K} U_{j,i_k}.$$

Given $\mathcal{A} \in \mathbb{R}^{m_1 \times \cdots \times m_K}$ and $m = \prod_{j=1}^{K} m_j$, let $\mathrm{vec}(\mathcal{A}) \in \mathbb{R}^m$ be vectorization of the matrix/tensor $\mathcal{A}$, $\mathrm{mat}_k(\mathcal{A}) \in \mathbb{R}^{m_k \times (m/m_k)}$ the mode-k matrix unfolding of $\mathcal{A}$, and $\mathrm{mat}_k(\mathrm{vec}(\mathcal{A})) = \mathrm{mat}_k(\mathcal{A})$.

2 A tensor factor model with a CP low-rank structure

Again, we specifically consider the following tensor factor model with CP low-rank structure (TFM-cp) for observations $\mathcal{X}_t \in \mathbb{R}^{d_1 \times \cdots \times d_K}$, $1 \le t \le T$,

$$\mathcal{X}_t = \sum_{i=1}^{r} w_i f_{it} \boldsymbol{a}_{i1} \otimes \boldsymbol{a}_{i2} \otimes \cdots \otimes \boldsymbol{a}_{iK} + \mathcal{E}_t,$$

where f_{it} is the unobserved latent factor process and $\boldsymbol{a}_{ik}$ are the fixed unknown factor loading vectors. We assume without loss of generality, $\mathbb{E}f_{it}^2 = 1$, $\|\boldsymbol{a}_{ik}\|_2 = 1$, for all $1 \le i \le r$ and $1 \le k \le K$. Then, all the signal strengths are contained in w_i. A key assumption of TFM-cp is that the factor process f_{it} is assumed to be uncorrelated across different factor processes, e.g. $\mathbb{E}f_{it-h}f_{jt} = 0$ for $i \ne j$ and $h \ge 1$. In addition, we assume that the noise tensor $\mathcal{E}_t$ are uncorrelated (white) across time, but with an arbitrary contemporary covariance structure, following Lam and Yao (2012) and R. Chen et al. (2022). In this paper, we consider the case that the order of the tensor K is fixed but the dimensions $d_1, \ldots, d_K \to \infty$ and rank r can be fixed or diverging.

Remark 1 By incorporating time, we may stack $\mathcal{X}_t$ into an order-$(K+1)$ tensor $\mathcal{Y} \in \mathbb{R}^{d_1 \times \cdots \times d_K \times T}$, with time t as the $(K+1)$th mode, referred to as the time-mode. Subsequently, model (1) can be reformulated as

$$\mathcal{Y} = \sum_{i=1}^{r} w_i \boldsymbol{a}_{i1} \otimes \boldsymbol{a}_{i2} \otimes \cdots \otimes \boldsymbol{a}_{iK} \otimes \boldsymbol{f}_i + \mathcal{E}, \tag{2}$$

where $\boldsymbol{f}_i = (f_{i1}, \ldots, f_{iT})^\top$. While it is enticing to directly estimate the signal part in (2) with standard tensor CP decomposition approaches based on the assumed CP structure, the dynamics and dependencies in the time direction (auto-dependency) are pivotal and warrant a distinct treatment. In our model, the component in the time direction is deemed latent and random. Consequently, it is crucial to examine the unique role of the time-mode and the (auto)-covariance structure in the time direction. The assumptions and interpretations inherent in our model, along with the corresponding estimation procedures and theoretical properties, markedly diverge from those of using the standard CP decompositions.

Remark 2 Ignoring the random noise $\mathcal{E}$, the CP decomposition in (2) is unique up to scaling and permutation indeterminacy if $\sum_{k=1}^{K} \mathcal{R}(\boldsymbol{A}_k) + \mathcal{R}(\boldsymbol{f}) \ge 2r + K$, where $\boldsymbol{A}_k = (\boldsymbol{a}_{1k}, \ldots, \boldsymbol{a}_{rk})$, $\boldsymbol{f} = (\boldsymbol{f}_1, \ldots, \boldsymbol{f}_r)$ and $\mathcal{R}(\boldsymbol{A}) = \max\{s : \text{any } s \text{ columns of the matrix } \boldsymbol{A} \text{ are linearly independent}\}$. Such a requirement provides a sufficient condition for uniqueness as per Kolda and Bader (2009). In the subsequent estimation procedure, we delve into the estimation of the autocovariance tensor $\boldsymbol{\Sigma}_h$ in (5). The sufficient identifiability condition for the CP decomposition of the autocovariance tensor becomes $2\sum_{k=1}^{K} \mathcal{R}(\boldsymbol{A}_k) \ge 2r + 2K - 1$. This

condition is significantly milder compared with the condition necessary to ensure statistical convergence.

Remark 3 (Comparison of TFM-cp with a Tucker low-rank structure). E. Y. Chen and Fan (2023), R. Chen et al. (2022), Y. Han et al. (2020, 2022) studied the following tensor factor models with a Tucker low-rank structure (TFM-tucker):

$$\mathcal{X}_t = \mathcal{F}_t \times_1 \boldsymbol{A}_1 \times \cdots \times \boldsymbol{A}_K + \mathcal{E}_t, \tag{3}$$

where the core tensor $\mathcal{F}_t \in \mathbb{R}^{r_1 \times \cdots \times r_K}$ is the latent factor process in a tensor form, and $\boldsymbol{A}_i$'s are $d_i \times r_i$ loading matrices. For example, when $K = 2$ (matrix time series), the TFM-cp can be rewritten as a TFM-tucker,

$$\boldsymbol{X}_t = \boldsymbol{A}_1 \boldsymbol{f}_t \boldsymbol{A}_2^\top + \boldsymbol{E}_t, \tag{4}$$

where $\boldsymbol{F}_t = \text{diag}(f_{1t}, \ldots, f_{rt})$, and $\boldsymbol{A}_1 = (\boldsymbol{a}_{11}, \ldots, \boldsymbol{a}_{r1})$ and $\boldsymbol{A}_2 = (\boldsymbol{a}_{12}, \ldots, \boldsymbol{a}_{r2})$ are matrices with the column vectors being $\boldsymbol{a}_{ik}$'s. There are four major differences between TFM-tucker and TFM-cp. First, TFM-tucker suffers from a severe identification problem, as the model remains equivalent if $\mathcal{F}_t$ is replaced by $\mathcal{F}_t \times_k \boldsymbol{R}$ and $\boldsymbol{A}_k$ replaced by $\boldsymbol{A}_k \boldsymbol{R}^{-1}$ for any invertible $r_k \times r_k$ matrix $\boldsymbol{R}$. For the $K = 2$ case, $\boldsymbol{X}_t = (\boldsymbol{A}_1 \boldsymbol{R}_1^{-1})(\boldsymbol{R}_1 \boldsymbol{f}_t \boldsymbol{R}_2^\top)(\boldsymbol{A}_2 \boldsymbol{R}_2^{-1})^\top + \boldsymbol{E}_t$ are all equivalent under TFM-tucker. Such ambiguity makes it difficult to find an 'optimal' representation of the model, which often leads to *ad hoc* and convenient representations that are difficult to interpret (Bai & Wang, 2014, 2015; Bekker, 1986; Neudecker, 1990). On the other hand, TFM-cp is uniquely defined up to sign changes, under an ordering of the signal strengths $w_1 \geq w_2 \geq \ldots \geq w_r$. As a result, the interpretation of the model becomes much easier. Second, although TFM-cp can be rewritten in the form of (3) with a *diagonal* core latent tensor consisting of the individual f_{it}'s, it is not under a typical Tucker form since TFM-tucker typically adopts the representation that the loading matrices $\boldsymbol{A}_k$'s are orthonormal, due to its identification problem. In TFM-cp, the loading vectors $\{\boldsymbol{a}_{ik}, 1 \leq i \leq r\}$ are not necessarily orthogonal vectors. In the $K = 2$ example in (4), if we find rotation matrices $\boldsymbol{R}_1$ and $\boldsymbol{R}_2$ so that $\boldsymbol{A}_1 \boldsymbol{R}_1^{-1}$ and $\boldsymbol{A}_2 \boldsymbol{R}_2^{-1}$ are orthonormal, then the corresponding core factor process in (4) becomes $\boldsymbol{R}_1 \boldsymbol{f}_t \boldsymbol{R}_2^\top$, no longer diagonal and with r^2 heavily correlated components, rather than r uncorrelated components. Third, TFM-cp separates the factor processes into a set of univariate time series, which enjoys great advantages over the tensor-valued factor processes in TFM-tucker. Modelling univariate time series are much easier and more flexible due to the vast repository of linear and nonlinear options. Lastly, TFM-cp is often much more parsimonious due to its restrictions, while enjoying great flexibility. Note that TFM-tucker is also a special case of TFM-cp, as it can be written as a sum of $r = r_1 \ldots r_K$ rank-one tensors, albeit with many repeated loading vectors. With its condensed formulation, in practical applications, the number of factors r needed under TFM-cp is typically much smaller than the total number $r_1 \ldots r_K$ of factors needed in TFM-tucker.

Remark 4 There are two different types of factor model assumptions in the literature. One type of factor models assumes that the common factors must have impact on 'most' (defined asymptotically) of the time series, but allows the idiosyncratic noise ($\mathcal{E}_t$) to have weak cross-correlations and weak autocorrelations; see, e.g. Forni et al. (2000), Bai and Ng (2002), Stock and Watson (2002), Fan et al. (2011, 2013), and E. Y. Chen and Fan (2023). PCA of the sample covariance

matrix is typically used to estimate the factor loading space, with various extensions. The other type of factor models assumes that the factors accommodate all dynamics, making the idiosyncratic noise 'white' with no autocorrelation, but allows substantial contemporary cross-correlation among the error process; see, e.g. Pena and Box (1987), Pan and Yao (2008), Lam et al. (2011), Lam and Yao (2012), and D. Wang et al. (2019). Under such assumptions, PCA is applied to the nonzero lagged autocovariance matrices. In this paper, we adopt the latter type of assumptions in our model development.

3 Estimation procedures

In this section, we focus on the estimation of the factors and loading vectors of model (1). The proposed procedure includes two steps: an initialization step using a new composite PCA (cPCA) procedure, presented in Algorithm 1, and an iterative refinement step using a new ISO procedure, presented in Algorithm 2. We call this two-step procedure HOPE (**H**igh-**O**rder **P**rojection **E**stimators) as it repeatedly perform high order projections on high order moments of the tensor observations. It utilizes the special structure of the model and leads to higher statistical and computational efficiency, which will be demonstrated later.

Algorithm 1 Initialization based on composite PCA (cPCA)

Input: The observations $\mathcal{X}_t \in \mathbb{R}^{d_1 \times \cdots \times d_K}$, $t = 1, ..., T$, the number of factors r, and the time lag h.

1: Evaluate $\widehat{\boldsymbol{\Sigma}}_h$ in (6), and unfold it to $d \times d$ matrix $\widehat{\boldsymbol{\Sigma}}_h^*$.

2: Obtain $\widehat{\boldsymbol{u}}_i$, $1 \le i \le r$, the top r eigenvectors of $(\widehat{\boldsymbol{\Sigma}}_h^* + \widehat{\boldsymbol{\Sigma}}_h^{*\top})/2$.

3: Compute $\widehat{\boldsymbol{a}}_{ik}^{\text{cpca}}$ as the top left singular vector of $\text{mat}_k(\widehat{\boldsymbol{u}}_i) \in \mathbb{R}^{d_k \times (d/d_k)}$, for all $1 \le k \le K$.

Output: $\widehat{\boldsymbol{a}}_{ik}^{\text{cpca}}$, $i = 1, ..., r$, $k = 1, ..., K$.

For $\mathcal{X}_t$ following (1), the lagged cross-product operator, denoted by $\boldsymbol{\Sigma}_h$, is the $(2K)$-tensor satisfying

$$\begin{aligned} \boldsymbol{\Sigma}_h &= \mathbb{E}\left[\sum_{t=h+1}^{T} \frac{\mathcal{X}_{t-h} \otimes \mathcal{X}_t}{T-h}\right] \\ &= \sum_{i=1}^{r} \lambda_{i,h} (\boldsymbol{a}_{i1} \otimes \boldsymbol{a}_{i2} \otimes \cdots \otimes \boldsymbol{a}_{iK})^{\otimes 2} \in \mathbb{R}^{d_1 \times \cdots \times d_K \times d_1 \times \cdots \times d_K}, \end{aligned} \tag{5}$$

for a given $h \ge 1$, where $\lambda_{i,h} = w_i^2 \mathbb{E} f_{i,t-h} f_{i,t}$. Note that the tensor $\boldsymbol{\Sigma}_h$ is expressed in a CP-decomposition form with each $\boldsymbol{a}_{ik}$ used twice. Let $\widehat{\boldsymbol{\Sigma}}_h$ be the sample version of $\boldsymbol{\Sigma}_h$,

$$\widehat{\boldsymbol{\Sigma}}_h = \sum_{t=h+1}^{T} \frac{\mathcal{X}_{t-h} \otimes \mathcal{X}_t}{T-h}. \tag{6}$$

When $\mathcal{X}_t$ is weakly stationary and $\mathcal{E}_t$ is white noise, a natural approach to estimating the loading vectors is via minimizing the empirical squared loss

$$(\boldsymbol{a}_{i1}, \boldsymbol{a}_{i2}, \ldots, \boldsymbol{a}_{iK}, 1 \le i \le r) = \underset{\substack{a_{i1}, a_{i2}, \ldots, a_{iK}, 1 \le i \le r, \\ \|\boldsymbol{a}_{i1}\|_2 = \ldots = \|\boldsymbol{a}_{iK}\|_2 = 1}}{\arg\min} \left\| \widehat{\boldsymbol{\Sigma}}_h - \sum_{i=1}^{r} \lambda_{i,h} (\boldsymbol{a}_{i1} \otimes \boldsymbol{a}_{i2} \otimes \cdots \otimes \boldsymbol{a}_{iK})^{\otimes 2} \right\|_{\text{HS}}^2, \tag{7}$$

Algorithm 2 Iterative Simultaneous Orthogonalization (ISO)

Input: The observations $\mathcal{X}_t \in \mathbb{R}^{d_1 \times \cdots \times d_K}$, $t = 1, \ldots, T$, the number of factors r, the warm-start initial estimates $\widehat{\boldsymbol{a}}_{ik}^{(0)}$, $1 \le i \le r$ and $1 \le k \le K$, the time lag h, the tolerance parameter $\epsilon > 0$, and the maximum number of iterations M.

1: Compute $\widehat{\boldsymbol{B}}_k^{(0)} = \widehat{\boldsymbol{A}}_k^{(0)} (\widehat{\boldsymbol{A}}_k^{(0)\top} \widehat{\boldsymbol{A}}_k^{(0)})^{-1} = (\widehat{\boldsymbol{b}}_{1k}^{(0)}, \ldots, \widehat{\boldsymbol{b}}_{rk}^{(0)})$ with $\widehat{\boldsymbol{A}}_k^{(0)} = (\widehat{\boldsymbol{a}}_{1k}^{(0)}, \ldots, \widehat{\boldsymbol{a}}_{rk}^{(0)}) \in \mathbb{R}^{d_k \times r}$ for $k = 1, \ldots, K$. Set $m = 0$.

2: **repeat**

3: Let $m = m + 1$.

4: **for** $k = 1$ to K.

5: **for** $i = 1$ to r.

6: Given previous estimates $\widehat{\boldsymbol{a}}_{ik}^{(m-1)}$, calculate

$$\mathcal{Z}_{t,ik}^{(m)} = \mathcal{X}_t \times_1 \widehat{\boldsymbol{b}}_{i1}^{(m)\top} \times_2 \cdots \times_{k-1} \widehat{\boldsymbol{b}}_{i,k-1}^{(m)\top} \times_{k+1} \widehat{\boldsymbol{b}}_{i,k+1}^{(m-1)\top} \times_{k+2} \cdots \times_K \widehat{\boldsymbol{b}}_{iK}^{(m-1)\top},$$

for $t = 1, \ldots, T$. Let

$$\widehat{\boldsymbol{\Sigma}}_h\left(\mathcal{Z}_{1:T,ik}^{(m)}\right) = \frac{1}{T-h} \sum_{t=h+1}^{T} \mathcal{Z}_{t-h,ik}^{(m)} \otimes \mathcal{Z}_{t,ik}^{(m)}.$$

Compute $\widehat{\boldsymbol{a}}_{ik}^{(m)}$ as the top eigenvector of $\widehat{\boldsymbol{\Sigma}}_h(\mathcal{Z}_{1:T,ik}^{(m)})/2 + \widehat{\boldsymbol{\Sigma}}_h(\mathcal{Z}_{1:T,ik}^{(m)})^\top/2$.

7: **end for**

8: Compute $\widehat{\boldsymbol{B}}_k^{(m)} = \widehat{\boldsymbol{A}}_k^{(m)} (\widehat{\boldsymbol{A}}_k^{(m)\top} \widehat{\boldsymbol{A}}_k^{(m)})^{-1} = (\widehat{\boldsymbol{b}}_{1k}^{(m)}, \ldots, \widehat{\boldsymbol{b}}_{rk}^{(m)})$ with $\widehat{\boldsymbol{A}}_k^{(m)} = (\widehat{\boldsymbol{a}}_{1k}^{(m)}, \ldots, \widehat{\boldsymbol{a}}_{rk}^{(m)})$.

9: **end for**

10: **until** $m = M$ or

$$\max_{1 \le i \le r} \max_{1 \le k \le K} \|\widehat{\boldsymbol{a}}_{ik}^{(m)} \widehat{\boldsymbol{a}}_{ik}^{(m)\top} - \widehat{\boldsymbol{a}}_{ik}^{(m-1)} \widehat{\boldsymbol{a}}_{ik}^{(m-1)\top}\|_{\mathrm{S}} \le \epsilon,$$

Output: Estimates

$$\begin{aligned}
\widehat{\boldsymbol{a}}_{ik}^{\mathrm{iso}} &= \widehat{\boldsymbol{a}}_{ik}^{(m)}, \quad i = 1, \ldots, r, \ k = 1, \ldots, K, \\
\widehat{w}_i^{\mathrm{iso}} &= \left(T^{-1} \sum_{t=1}^{T} \left(\mathcal{X}_t \times_{k=1}^{K} \widehat{\boldsymbol{b}}_{ik}^{(m)\top} \right)^2 \right)^{1/2}, \quad i = 1, \ldots, r, \\
\widehat{f}_{it}^{\mathrm{iso}} &= \left(\widehat{w}_i^{\mathrm{iso}}\right)^{-1} \cdot \mathcal{X}_t \times_{k=1}^{K} \widehat{\boldsymbol{b}}_{ik}^{(m)\top}, \quad i = 1, \ldots, r, \ t = 1, \ldots, T, \\
\widehat{\mathcal{X}}_t^{\mathrm{iso}} &= \sum_{i=1}^{r} \mathcal{X}_t \times_{k=1}^{K} \widehat{\boldsymbol{b}}_{ik}^{(m)\top} \times_{k=1}^{K} \widehat{\boldsymbol{a}}_{ik}^{(m)}, \quad t = 1, \ldots, T.
\end{aligned}$$

where the Hilbert Schmidt norm for a tensor $\mathcal{A}$ is defined as $\|\mathcal{A}\|_{\mathrm{HS}} = \|\mathrm{vec}(\mathcal{A})\|_2$. In other words, $\boldsymbol{a}_{i1} \otimes \boldsymbol{a}_{i2} \otimes \cdots \otimes \boldsymbol{a}_{iK}$ can be estimated by the leading *principal component* of the sample autocovariance tensor $\widehat{\boldsymbol{\Sigma}}_h$. However, due to the nonconvexity of (7) or its variants, a straightforward implementation of many local search algorithms, such as gradient descent and alternating minimization, may easily get trapped into local optima and result in suboptimal statistical performance. As shown by Auffinger et al. (2013), there could be an exponential number of local optima and the great majority of these local optima are far from the best low-rank approximation. However, if we start from an appropriate initialization not too far from the global optimum, then

a local optimum reached may be as good an estimator as the global optimum. A critical task in estimating the factor loading vectors is thus to obtain good initialization.

We develop a warm initialization procedure, the composite PCA (cPCA) procedure. Note that, if we unfold $\boldsymbol{\Sigma}_b$ into a $d \times d$ matrix $\boldsymbol{\Sigma}_b^*$, where $d = d_1 \ldots d_K$, then (5) implies that

$$\boldsymbol{\Sigma}_b^* = \sum_{i=1}^{r} \lambda_{i,b} \boldsymbol{a}_i \boldsymbol{a}_i^\top, \tag{8}$$

a sum of r rank-one matrices, each of the form $\boldsymbol{a}_i \boldsymbol{a}_i^\top$, where $\boldsymbol{a}_i = \mathrm{vec}(\bigotimes_{k=1}^K \boldsymbol{a}_{ik})$. This is very close to the principal component decomposition of $\boldsymbol{\Sigma}_b^*$, except that $\boldsymbol{a}_i$'s are not necessarily orthogonal in this case. However, the following intuition provides a solid justification of using PCA to obtain an estimate of $\boldsymbol{a}_i$. We call this estimator the cPCA estimator.

The accuracy of using the principal components of $\boldsymbol{\Sigma}_b^*$ as the estimate of $\boldsymbol{a}_i$ heavily depends on the coherence of the components, defined as $\vartheta = \max_{1 \le i < j \le r} |\boldsymbol{a}_i^\top \boldsymbol{a}_j|$, the maximum pairwise correlation among the $\boldsymbol{a}_i$'s. When the components are orthogonal ($\vartheta = 0$), there is no error in using PCA. The main idea of cPCA is to take advantage of the special structure of TFM-cp, which leads to a multiplicative high-order coherence of the CP components. In the following, we provide an analysis of ϑ under TFM-cp.

Let $\boldsymbol{A}_k = (\boldsymbol{a}_{1k}, \ldots, \boldsymbol{a}_{rk}) \in \mathbb{R}^{d_k \times r}$ be the matrix with $\boldsymbol{a}_{ik}$ as its columns, and $\boldsymbol{A}_k^\top \boldsymbol{A}_k = (\sigma_{ij,k})_{r \times r}$. As $\sigma_{ii,k} = \|\boldsymbol{a}_{ik}\|_2^2 = 1$, the correlation among columns of $\boldsymbol{A}_k$ can be measured by

$$\vartheta_k = \max_{1 \le i < j \le r} |\sigma_{ij,k}|, \quad \delta_k = \|\boldsymbol{A}_k^\top \boldsymbol{A}_k - I_r\|_{\mathrm{S}}, \quad \eta_{jk} = \left(\sum_{i \in [r] \setminus \{j\}} \sigma_{ij,k}^2 \right)^{1/2}. \tag{9}$$

Similarly, we use

$$\vartheta = \max_{1 \le i < j \le r} |\boldsymbol{a}_i^\top \boldsymbol{a}_j|, \quad \delta = \|\boldsymbol{A}^\top \boldsymbol{A} - I_r\|_{\mathrm{S}}, \tag{10}$$

to measure the correlation of the matrix $\boldsymbol{A} = (\boldsymbol{a}_1, \ldots, \boldsymbol{a}_r) \in \mathbb{R}^{d \times r}$ with $\boldsymbol{a}_i = \mathrm{vec}(\bigotimes_{k=1}^K \boldsymbol{a}_{ik})$ and $d = \prod_{k=1}^K d_k$. It can be seen that the coherence ϑ has the bound $\vartheta \le \prod_{k=1}^K \vartheta_k \le \vartheta_{\max}^K$, due to $\boldsymbol{a}_i^\top \boldsymbol{a}_j = \prod_{k=1}^K \boldsymbol{a}_{ik}^\top \boldsymbol{a}_{jk} = \prod_{k=1}^K \sigma_{ij,k}$. The spectrum norm δ is also bounded by the multiplicative of correlation measures in (9). More specifically, we have the following proposition.

Proposition 1 Define $\mu_* = \max_j \min_{k_1,k_2} \max_{i \ne j} \prod_{k \ne k_1, k \ne k_2, k \in [K]} \sqrt{r} |\sigma_{ij,k}| / \eta_{jk} \in [1, r^{K/2-1}]$ as the (leave-two-out) mutual coherence of $\boldsymbol{A}_1, \ldots, \boldsymbol{A}_K$. Then, $\delta \le \min_{1 \le k \le K} \delta_k$ and

$$\delta \le (r-1)\vartheta, \quad \text{and} \quad \vartheta \le \prod_{k=1}^K \vartheta_k \le \vartheta_{\max}^K, \tag{11}$$

$$\delta \le \mu_* r^{1-K/2} \max_{j \le r} \prod_{k=1}^K \eta_{jk} \le \mu_* r^{1-K/2} \prod_{k=1}^K \delta_k. \tag{12}$$

When (most of) the quantities in (9) are small, the products in (11) would be very small so that the $\boldsymbol{a}_i$'s are nearly orthogonal. For example, if $(\boldsymbol{a}_{11}, \boldsymbol{a}_{21})$ and $(\boldsymbol{a}_{12}, \boldsymbol{a}_{22})$ both have i.i.d. bi-variate random rows with correlation coefficients ρ_1 and ρ_2, and independent, then the population correlation coefficient of $\mathrm{vec}(\boldsymbol{a}_{11} \otimes \boldsymbol{a}_{12})$ and $\mathrm{vec}(\boldsymbol{a}_{21} \otimes \boldsymbol{a}_{22})$ is $\rho_1 \rho_2$, though the variation of the sample correlation coefficient depends on the length of the $\boldsymbol{a}_{ik}$'s.

Remark 5 Let polylog denote the polynomial of the logarithm. The incoherence condition such as $\vartheta_{\max} \lesssim \text{ploylog}(d_{\min})/\sqrt{d_{\min}}$ is commonly imposed in the literature for generic CP decomposition; see e.g. Anandkumar et al. (2014b, 2014c), Sun et al. (2017), and Hao et al. (2020). Proposition 1 establishes a connection between δ, δ_k and the ϑ_k in the same framework of incoherence considerations. The parameters δ_k and δ quantify the nonorthogonality of the factor loading vectors, and play a key role in our theoretical analysis, as the performance bound of cPCA estimators involves δ. Differently from the existing literature depending on $\vartheta_{\max}$, the cPCA exploits δ or the much smaller ϑ (comparing to $\vartheta_{\max}$), thus has better properties when $K \geq 2$. Note that the idea of using tensor unfolding to enhance incoherence can be traced back to Huang et al. (2015), Jain and Oh (2014), and Allman et al. (2009), though their incoherence measure is slightly different from ours. The most notable advances from these studies is that we establish a nonasymptotic bound for the estimated loading vectors in the presence of noise (c.f. Theorem 1).

The pseudo-code of cPCA is provided in Algorithm 1. Though $\boldsymbol{\Sigma}_b^*$ is symmetric, its sample version $\widehat{\boldsymbol{\Sigma}}_b^*$ in general is not. We use $(\widehat{\boldsymbol{\Sigma}}_b^* + \widehat{\boldsymbol{\Sigma}}_b^{*\top})/2$ to ensure symmetry and reduce the noise. The cPCA produces definitive initialization vectors up to the sign change.

After obtaining a warm start via cPCA (Algorithm 1), we engage an ISO algorithm (Algorithm 2) to refine the solution of $\boldsymbol{a}_{ik}$ and obtain estimations of the factor process f_{it} and the signal strength w_i. Algorithm 2 can be viewed as an extension of HOOI (De Lathauwer et al., 2000; Zhang & Xia, 2018) and the iterative projection algorithm in Y. Han et al. (2020) to undercomplete ($r < d_{\min}$) and nonorthogonal CP decompositions. It is motivated by the following observation. Define $\boldsymbol{A}_k = (\boldsymbol{a}_{1k}, \ldots, \boldsymbol{a}_{rk})$ and $\boldsymbol{B}_k = \boldsymbol{A}_k(\boldsymbol{A}_k^\top \boldsymbol{A}_k)^{-1} = (\boldsymbol{b}_{1k}, \ldots, \boldsymbol{b}_{rk}) \in \mathbb{R}^{d_k \times r}$. Let

$$\mathcal{Z}_{t,ik} = \mathcal{X}_t \times_1 \boldsymbol{b}_{i1}^\top \times_2 \cdots \times_{k-1} \boldsymbol{b}_{i,k-1}^\top \times_{k+1} \boldsymbol{b}_{i,k+1}^\top \times_{k+2} \cdots \times_K \boldsymbol{b}_{iK}^\top, \tag{13}$$

$$\mathcal{E}_{t,ik}^* = \mathcal{E}_t \times_1 \boldsymbol{b}_{i1}^\top \times_2 \cdots \times_{k-1} \boldsymbol{b}_{i,k-1}^\top \times_{k+1} \boldsymbol{b}_{i,k+1}^\top \times_{k+2} \cdots \times_K \boldsymbol{b}_{iK}^\top. \tag{14}$$

Since $\boldsymbol{a}_{jk}^\top \boldsymbol{b}_{ik} = I_{\{i=j\}}$, model (1) implies that

$$\mathcal{Z}_{t,ik} = w_i f_{it} \boldsymbol{a}_{ik} + \mathcal{E}_{t,ik}^*. \tag{15}$$

Here $\mathcal{Z}_{t,ik}$ is a vector, and (15) is in a factor model form with a univariate factor. The estimation of $\boldsymbol{a}_{ik}$ can be done easily and much more accurately than dealing with the much larger $\mathcal{X}_t$. The operation in (13) achieves two objectives. First, by multiplying a vector on every mode except the kth mode to $\mathcal{X}_t$, it reduces the tensor to a vector. It also serves as an averaging operation to reduce the noise variation. Second, as $\boldsymbol{b}_{ik}$ is orthogonal to all $\boldsymbol{a}_{jk}$ except $\boldsymbol{a}_{ik}$, it is an orthogonal projection operation that eliminates all $\otimes_{k=1}^K \boldsymbol{a}_{jk}$ terms in (1) except the ith term, resulting in (15). If the matrix $\boldsymbol{A}_k^\top \boldsymbol{A}_k$ is not ill-conditioned, i.e. $\{\boldsymbol{a}_{ik}, 1 \leq i \leq r\}$ are not highly correlated, then $\boldsymbol{B}_k$ and all individual $\boldsymbol{b}_{jk}$ are well defined and this procedure shall work well. Under proper conditions on the combined noise tensor $\mathcal{E}_{t,ik}^*$, estimation of the loading vectors $\boldsymbol{a}_{ik}$ based on $\mathcal{Z}_{t,ik}$ can be made significantly more accurate, as the statistical error rate now depends on d_k rather than $d_1 d_2 \ldots d_k$. Intuitively, $\boldsymbol{b}_{ik}$ can also be viewed as a form of (normalized) residuals of $\boldsymbol{a}_{ik}$ projected onto the space spanned by $\{\boldsymbol{a}_{jk}, j \neq i, 1 \leq j \leq r\}$.

In practice, we do not know $\boldsymbol{b}_{il}$, for $1 \leq i \leq r$, $1 \leq l \leq K$ and $l \neq k$. Similar to back-fitting algorithms, we iteratively estimate the loading vector $\boldsymbol{a}_{ik}$ at iteration number m based on

$$\mathcal{Z}_{t,ik}^{(m)} = \mathcal{X}_t \times_1 \widehat{\boldsymbol{b}}_{i1}^{(m)\top} \times_2 \cdots \times_{k-1} \widehat{\boldsymbol{b}}_{i,k-1}^{(m)\top} \times_{k+1} \widehat{\boldsymbol{b}}_{i,k+1}^{(m-1)\top} \times_{k+2} \cdots \times_K \widehat{\boldsymbol{b}}_{iK}^{(m-1)\top},$$

using the estimate $\widehat{\boldsymbol{b}}_{il}^{(m-1)}$, $k < l \le K$, obtained in the previous iteration and the estimate $\widehat{\boldsymbol{b}}_{il}^{(m)}$, $1 \le l < k$, obtained in the current iteration. As we shall show in the next section, such an iterative procedure leads to a much improved statistical rate in the high-dimensional tensor factor model scenarios, as if all $\boldsymbol{b}_{il}$, $1 \le i \le r$, $1 \le l \le K$, $l \ne k$, are known and we indeed observe $\mathcal{Z}_{t,ik}$ that follows model (15). Note that the projection error is

$$\mathcal{Z}_{t,ik}^{(m)} - \mathcal{Z}_{t,ik} = \sum_{j=1}^{r} w_j f_{j,t} \xi_{ij}^{(m)} \boldsymbol{a}_{jk} + \mathcal{E}_{t,ik}^{*(m)} - \mathcal{E}_{t,ik}^{*},$$

where

$$\xi_{ij}^{(m)} = \prod_{\ell=1}^{k-1} [\boldsymbol{a}_{j\ell}^{\top} \widehat{\boldsymbol{b}}_{i\ell}^{(m)}] \prod_{\ell=k+1}^{K} [\boldsymbol{a}_{j\ell}^{\top} \widehat{\boldsymbol{b}}_{i\ell}^{(m-1)}] - I_{\{i=j\}}, \tag{16}$$

and $\mathcal{E}_{t,ik}^{*(m)}$ is that in (14) with $\boldsymbol{b}_{ik}$ replaced with $\widehat{\boldsymbol{b}}_{ik}^{(m-1)}$ or $\widehat{\boldsymbol{b}}_{ik}^{(m)}$. The multiplicative measure of projection error $|\xi_{ij}^{(m)}|$ decays rapidly since, for $j \ne i$, $\boldsymbol{a}_{j\ell}^{\top} \widehat{\boldsymbol{b}}_{i\ell}^{(m)}$ goes to zero quickly as the iteration m increases, and $\xi_{ij}^{(m)}$ is a product of $K-1$ such terms. In fact, the higher the tensor order K is, the faster the error goes to zero.

Remark 6 (Comparison with alternating least square). The updates in Algorithm 2 can be viewed as a variant of the standard alternating least-squares procedure. For example, suppose that $\widehat{\boldsymbol{b}}_{ik}^{(m-1)}$, $1 \le i \le r$, $2 \le k \le K$, are fixed. Then the optimization problem to update $\boldsymbol{a}_{i1}^{(m)}$ for each $1 \le i \le r$ can be rewritten as

$$\arg \min_{\boldsymbol{a}_{i1} \in \mathbb{R}^{d_1}} \left\| \widehat{\boldsymbol{\Sigma}}_h \times_{k=2}^{K} \widehat{\boldsymbol{b}}_{ik}^{(m-1)} \times_{k=K+2}^{2K} \widehat{\boldsymbol{b}}_{i,k-K}^{(m-1)} - w_i \boldsymbol{a}_{i1} \boldsymbol{a}_{i1}^{\top} \right\|_{\mathrm{F}}.$$

This is a least-squares problem. However, the algorithm cannot be viewed as an alternating least-square procedure since we do not have an over-arching (least-square) objective function such that every iteration is done to minimize the objective function given other components. This is due to the construction and involvement of $\boldsymbol{b}_{ik}$ in the algorithm. As a matter of fact, if one uses standard ALS to minimize the objective function in (7), to update mode k, it would involve the inverse of the Hadamard product of $\boldsymbol{A}_{k'}^{\top} \boldsymbol{A}_{k'}$ for $k' \ne k$. In contrast, due to the nice property of $\mathcal{Z}_{t,ik}$ (defined based on $\boldsymbol{B}_{k'}$ for $k' \ne k$), we only need to compute the inverse of $\boldsymbol{A}_{k'}^{\top} \boldsymbol{A}_{k'}$ for each $k' \ne k$, not their Hadamard product.

Remark 7 (The role of h). In Algorithm 1, we use a fixed $h \ge 1$. Let $\widehat{\lambda}_{1,h} \ge \widehat{\lambda}_{2,h} \ge \cdots \ge \widehat{\lambda}_{d,h}$ be the eigenvalues of $\widetilde{\boldsymbol{\Sigma}}_h^{*} := (\widehat{\boldsymbol{\Sigma}}_h^{*} + \widehat{\boldsymbol{\Sigma}}_h^{*\top})/2$. In practice, we may select h to maximize the fraction of the explained variance $\sum_{i=1}^{r} \widehat{\lambda}_{i,h}^2 / \sum_{i=1}^{d} \widehat{\lambda}_{i,h}^2$ under different lag values $1 \le h \le h_0$, given some pre-specified maximum allowed lag h_0. Step 2 in Algorithm 1 can be improved by accumulating information from different time lags. For example, let $\widehat{\boldsymbol{U}}^{(0)} \in R^{d \times r}$ be a matrix with its columns $\widehat{\boldsymbol{u}}_i$'s being the top r eigenvectors of $\widetilde{\boldsymbol{\Sigma}}_h^{*}$. With such $\widehat{\boldsymbol{U}}^{(0)}$ as the initialization, we may iteratively refine $\widehat{\boldsymbol{U}}^{(m)}$ to be the top r eigenvectors of $\sum_{h=1}^{h_0} \widetilde{\boldsymbol{\Sigma}}_h^{*} \widehat{\boldsymbol{U}}^{(m-1)} \widehat{\boldsymbol{U}}^{(m-1)\top} \widetilde{\boldsymbol{\Sigma}}_h^{*}$.

Remark 8 (Condition number of $\widehat{\boldsymbol{A}}_k^{(m)\top} \widehat{\boldsymbol{A}}_k^{(m)}$). Our theoretical analysis assumes that the condition number of the matrix $\boldsymbol{A}_k^{\top} \boldsymbol{A}_k$ is bounded. However, in practice, the condition number of $\widehat{\boldsymbol{A}}_k^{(m)\top} \widehat{\boldsymbol{A}}_k^{(m)}$ in Algorithm 2 may be very large, especially

when $m = 0$. We suggest a simple regularized strategy. Define the eigen decomposition $\widehat{\boldsymbol{A}}_k^{(m)\top}\widehat{\boldsymbol{A}}_k^{(m)} = \boldsymbol{V}_k^{(m)}\boldsymbol{\Lambda}_k^{(m)}\boldsymbol{V}_k^{(m)\top}$. For all eigenvalues in $\boldsymbol{\Lambda}_k^{(m)}$ that are smaller than a numeric constant c (e.g. $c = 0.1$), we set them to c. Denote the resulting matrix as $\widetilde{\boldsymbol{\Lambda}}_k^{(m)}$ and get the corresponding $\widehat{\boldsymbol{B}}_k^{(m)}$ by $\widehat{\boldsymbol{A}}_k^{(m)\top}(\boldsymbol{V}_k^{(m)}\widetilde{\boldsymbol{\Lambda}}_k^{(m)}\boldsymbol{V}_k^{(m)\top})^{-1}$. Many alternative empirical methods can also be applied to bound the condition number.

Remark 9 Algorithm 1 requires that $\delta < 1$ in order to obtain reasonable estimates. And it can accommodate the case that $r \geq d_{\max}$. In contrast, Algorithm 2 needs stronger conditions that $\delta_k < 1$ and $r \leq d_{\min}$ to rule out the possibility of colinearity, as $\boldsymbol{A}_k^\top\boldsymbol{A}_k$ needs to be invertible. It may not hold under certain situations. For example, $\boldsymbol{a}_{1k} = \boldsymbol{a}_{2k}$ would lead to an ill-conditioned $\boldsymbol{A}_k^\top\boldsymbol{A}_k$. In such cases, the incoherence condition commonly required in the literature, e.g. $\vartheta_{\max} \ll 1$, is also violated. It is possible to extend our approach to a more sophisticated projection scheme so the conditions can be weakened. As it requires more sophisticated analysis both on the methodology and on the theory, we do not purse this direction in this paper.

Remark 10 As mentioned before, $\boldsymbol{a}_{i1} \otimes \boldsymbol{a}_{i2} \otimes \cdots \otimes \boldsymbol{a}_{iK}$ can be regarded as the *principal component* of the autocovariance tensor $\boldsymbol{\Sigma}_h$. Hence, our HOPE estimators (Algorithms 1 and 2 together) can also be characterized as a procedure of *principal component analysis* for order $2K$ autocovariance tensor, albeit with a special structure in (5).

Remark 11 (The number of factors). Here the estimators are constructed with given rank r, though in the theoretical analysis it is allowed to diverge. Determining the number of factors in a data-driven way has been an important research topic in the factor model literature. Bai and Ng (2002, 2007), and Hallin and Liška (2007) proposed consistent estimators in the vector factor models based on the information criteria approach. Lam and Yao (2012) and Ahn and Horenstein (2013) developed an alternative approach to study the ratio of each pair of adjacent eigenvalues. Recently, Y. Han et al. (2022) established a class of rank determination approaches for the factor models with Tucker low-rank structure, based on both the information criterion and the eigenratio criterion. Those procedures can be extended to TFM-cp.

4 Theoretical properties

In this section, we shall investigate the statistical properties of the proposed algorithms described in the last section. Our theories provide theoretical guarantees for consistency and present statistical error rates in the estimation of the factor loading vectors $\boldsymbol{a}_{ik}$, $1 \leq i \leq r$, $1 \leq k \leq K$, under proper regularity conditions. As the loading vector $\boldsymbol{a}_{ik}$ is identifiable only up to the sign change, we use

$$\|\widehat{\boldsymbol{a}}_{ik}\widehat{\boldsymbol{a}}_{ik}^\top - \boldsymbol{a}_{ik}\boldsymbol{a}_{ik}^\top\|_{\mathrm{S}} = \sqrt{1 - (\widehat{\boldsymbol{a}}_{ik}^\top\boldsymbol{a}_{ik})^2} = \sup_{z \perp \boldsymbol{a}_{ik}} |\boldsymbol{z}^\top\widehat{\boldsymbol{a}}_{ik}|$$

to measure the distance between $\widehat{\boldsymbol{a}}_{ik}$ and $\boldsymbol{a}_{ik}$. Recall $\boldsymbol{\Sigma}_h = \mathbb{E}\widehat{\boldsymbol{\Sigma}}_h = \sum_{i=1}^r \lambda_{i,h}(\boldsymbol{a}_{i1} \otimes \boldsymbol{a}_{i2} \otimes \cdots \otimes \boldsymbol{a}_{iK})^{\otimes 2}$, as in (5) and $\lambda_{i,h} = w_i^2\mathbb{E}f_{i,t-h}f_{i,t}$. We will also continue to use the notations $\boldsymbol{A}_k = (\boldsymbol{a}_{1k}, \ldots, \boldsymbol{a}_{rk}) \in \mathbb{R}^{d_k \times r}$, and $\boldsymbol{B}_k = \boldsymbol{A}_k(\boldsymbol{A}_k^\top\boldsymbol{A}_k)^{-1} = (\boldsymbol{b}_{1k}, \ldots, \boldsymbol{b}_{rk}) \in \mathbb{R}^{d_k \times r}$. Let $d = \prod_{k=1}^K d_k$, $d_{\min} = \min\{d_1, \ldots, d_K\}$, $d_{\max} = \max\{d_1, \ldots, d_K\}$ and $d_{-k} = \prod_{j \neq k} d_j$.

To present theoretical properties of the proposed procedures, we impose the following assumptions.

Assumption 1 The error process $\mathcal{E}_t$ are independent Gaussian tensors, conditioning on the factor process $\{f_{it}, 1 \le i \le r, t \in \mathbb{Z}\}$. In addition, there exists some constant $\sigma > 0$, such that

$$\mathbb{E}(u^\top \mathrm{vec}(\mathcal{E}_t))^2 \le \sigma^2 \|u\|_2^2, \quad u \in \mathbb{R}^d.$$

Assumption 2 Assume the factor process f_{it}, $1 \le i \le r$ is stationary and strong α-mixing in t, with $\mathbb{E}f_{it}^2 = 1$, $\mathbb{E}f_{it-h}f_{it} \ne 0$, $\mathbb{E}f_{it-h}f_{jt} = 0$ for all $i \ne j$ and $h \ge 1$. Let $F_t = (f_{1t}, \ldots, f_{rt})^\top$. For any $v \in \mathbb{R}^r$ with $\|v\|_2 = 1$,

$$\max_t \mathbb{P}\left(\left|v^\top F_t\right| \ge x\right) \le c_1 \exp(-c_2 x^{\gamma_2}), \tag{17}$$

where c_1, c_2 are some positive constants and $0 < \gamma_2 \le 2$. In addition, the mixing coefficient satisfies

$$\alpha(m) \le \exp(-c_0 m^{\gamma_1}) \tag{18}$$

for some constant $c_0 > 0$ and $0 < \gamma_1 \le 1$, where

$$\alpha(m) = \sup_t \Big\{ \Big| \mathbb{P}(A \cap B) - \mathbb{P}(A)\mathbb{P}(B) \Big| : A \in \sigma(f_{is}, 1 \le i \le r, s \le t), B \in \sigma(f_{is}, 1 \le i \le r, s \ge t + m) \Big\}.$$

Assumption 3 Assume $h \le T/4$ is fixed, and $\lambda_{1,h}, \ldots, \lambda_{r,h}$ are all distinct. Without loss of generality, let $\lambda_{1,h} > \lambda_{2,h} > \cdots > \lambda_{r,h} > 0$. Here, we emphasize that $\lambda_{i,h}$ depends on h, though in other places when h is fixed we will omit h in the notation.

Assumption 1 is similar to those on the noise imposed in Lam et al. (2011), Lam and Yao (2012), and Y. Han et al. (2020). It accommodates general patterns of dependence among individual time series fibres, but also allows a presentation of the main results with manageable analytical complexity. The normality assumption, which ensures fast statistical error rates in our analysis, is imposed for technical convenience. In theory, it can be supplanted by a sub-Gaussian condition, or replaced with more general distributions with heavier tails. However, adopting such weaker conditions would significantly complicate the formulae, statistical outcomes, and requisite conditions within our time series framework. This added complexity would likely detract from the paper's readability, without providing additional statistical insights. Our primary goal is to maintain the readers' focus on the core content of the paper, and thus, we have chosen to assume additive Gaussian errors. This choice simplifies the exposition without compromising the fundamental tenets and the findings of our study.

Assumption 2 is standard. It allows a very general class of time series models, including causal ARMA processes with continuously distributed innovations; see also Tong (1990), Bradley (2005), Tsay (2005), Fan and Yao (2003), Rosenblatt (2012), and Tsay and Chen (2018), among others. The restriction $\gamma_1 \le 1$ is introduced only for presentation convenience. Assumption 2 requires that the tail probability of f_{it} decays exponentially fast. In particular, when $\gamma_2 = 2$, f_{it} is sub-Gaussian.

Assumption 3 is sufficient to guarantee that all the factor loading vectors $\boldsymbol{a}_{ik}$ can be uniquely identified up to the sign change. The parameters λ_i can be viewed as an analogue of eigenvalues in the order-$2K$ tensor $\boldsymbol{\Sigma}_h$. Similar to the eigen decomposition of a matrix, if some λ_i are equal, estimation of the loading vectors $\boldsymbol{a}_{ik}$ may suffer from label shift across i. When h is fixed and $\mathbb{E}f_{i,t}f_{i,t-h} \asymp 1$, $\lambda_i \asymp w_i^2$. The signal strength of each factor is measured by λ_i.

Let us first study the behaviour of the cPCA estimators in Algorithm 1. Theorem 1 presents the performance bounds, which depends on the coherence (the degree of nonorthogonality) of the factor loading vectors.

Let

$$\lambda_* = \min_{1\le i\le r+1} \{\lambda_{i-1} - \lambda_i\} \tag{19}$$

with $\lambda_0 = \infty$, $\lambda_{r+1} = 0$, be the minimum gap between the signal strengths of the factors.

Theorem 1 Suppose Assumptions 1, 2, 3 hold. Let $1/\gamma = 1/\gamma_1 + 2/\gamma_2$, $h \le T/4$, and $\delta < 1$ with δ defined in (10). In an event with probability at least $1 - (Tr)^{-C_1} - e^{-d}$, the following error bound holds for the estimation of the loading vectors $\boldsymbol{a}_{ik}$ using Algorithm 1 (cPCA).

$$\|\widehat{\boldsymbol{a}}_{ik}^{\text{cpca}}\widehat{\boldsymbol{a}}_{ik}^{\text{cpca}\top} - \boldsymbol{a}_{ik}\boldsymbol{a}_{ik}^\top\|_{\text{S}} \le \left(1 + \frac{2\lambda_1}{\lambda_*}\right)\delta + \frac{C_2 R^{(0)}}{\lambda_*}, \tag{20}$$

for all $1 \le i \le r$, $1 \le k \le K$, where C_1, C_2 are some positive constants, and

$$R^{(0)} = \max_{1\le i\le r} w_i^2 \left(\sqrt{\frac{r+\log T}{T}} + \frac{(r+\log T)^{1/\gamma}}{T}\right) + \sigma^2\sqrt{\frac{d}{T}} + \sigma \max_{1\le i\le r} w_i \sqrt{\frac{d}{T}}. \tag{21}$$

Remark 12 We note that the eigengap λ_* in Algorithm 1 (cPCA) is not a requisite for the iterative Algorithm 2 (ISO). Algorithm 1 (cPCA) necessitates a significant separation among the different singular values λ_i's. This condition can be relaxed through the use of random slicing (Anandkumar et al., 2014c; Auddy & Yuan, 2023a), a widely recognized method for initialization in tensor CP decomposition. In our framework, the core step of random slicing is the construction of a projected autocovariance tensor $\boldsymbol{\Sigma}_h \times_k g_k \times_{K+k} \tilde{g}_k$, where g_k and $\tilde{g}_k$ are independently generated Gaussian random vectors. Due to the randomness of g_k and $\tilde{g}_k$, even if all λ_i's are equal, the singular values of $\boldsymbol{\Sigma}_h \times_k g_k \times_{K+k} \tilde{g}_k$ will inherently differ. Furthermore, we can generate a sufficiently large eigengap between the top two singular values of the projected autocovariance tensor through multiple rounds of random slicing, so that the leading component of the projected autocovariance tensor is identifiable. Since $\boldsymbol{\Sigma}_h \times_k g_k \times_{K+k} \tilde{g}_k$ is an order $2K-2$ tensor, we can still employ cPCA and utilize the benefits of higher-order coherence in Proposition 1 when $K > 2$. The existing results in Theorem 1 can be extended to such settings, although it would require more sophisticated theoretical analysis.

The first term in the upper bound (20) is induced by the nonorthogonality of the loading vectors $\boldsymbol{a}_{ik}$, which can be viewed as bias. The second term in (20) comes from a concentration bound for the random noise, and thus can be interpreted as stochastic error. By Proposition 1, it implies that a larger K (e.g. higher order tensors) leads to smaller bias and higher statistical accuracy of cPCA. If $\delta \gtrsim R^{(0)}/\lambda_1$, then the error bound (20) is dominated by the bias related to δ, otherwise it is dominated by the stochastic error. Equation (21) shows that $R^{(0)}$ in the stochastic error comes from the fluctuation of the factor process f_{it} (the first two terms) and the noise $\mathcal{E}_t$ in (1) (the other two terms). When $\sum_{i=1}^r \lambda_i \asymp r\lambda_1 \asymp rw_1^2$ and $R^{(0)}/\lambda_1 + T/d \lesssim 1$, the terms related to the noise becomes $\sqrt{r/T}/\sqrt{\text{SNR}}$, where the signal-to-noise ratio (SNR) is

$$\text{SNR} := \mathbb{E}\left\|\sum_{i=1}^r w_i f_{it} \otimes_{k=1}^K \boldsymbol{a}_{ik}\right\|_{\text{HS}}^2 / \mathbb{E}\|\mathcal{E}_t\|_{\text{HS}}^2 = \sum_{i=1}^r w_i^2/(\sigma^2 d) \asymp r\lambda_1/(\sigma^2 d). \tag{22}$$

Roughly speaking though not completely correct, the term $\lambda_i - \lambda_{i+1}$ can be viewed as the gap of ith and $(i+1)$th largest eigenvalues of $\boldsymbol{\Sigma}_b^*$ with $\boldsymbol{\Sigma}_b^*$ given in (8). In particular, if $\lambda_1 \asymp \cdots \asymp \lambda_r \asymp w_1^2$, then $\lambda_* \asymp w_1^2/r$. In this case, the bound (20) can be simplified to

$$\begin{aligned}&\|\widehat{\boldsymbol{a}}_{ik}^{\mathrm{cpca}}\widehat{\boldsymbol{a}}_{ik}^{\mathrm{cpca}\top} - \boldsymbol{a}_{ik}\boldsymbol{a}_{ik}^\top\|_{\mathrm{S}} \\ &\quad\le C_3 r\delta + C_4 r\left(\sqrt{\frac{r+\log T}{T}} + \frac{(r+\log T)^{1/\gamma}}{T}\right) + \frac{C_4\sigma^2 r\sqrt{d}}{w_1^2\sqrt{T}} + \frac{C_4\sigma r\sqrt{d}}{w_1\sqrt{T}}.\end{aligned} \tag{23}$$

Then, by (23) and Proposition 1, the consistency of the cPCA estimators only requires the incoherence parameter to be at most $\vartheta_{\max} \lesssim r^{-2/K}$.

Next, let us consider the statistical performance of the iterative algorithm (Algorithm 2) after cPCA initialization, i.e. HOPE estimators. As discussed earlier, the operation in (13) achieves dimension reduction by projecting $\mathcal{X}_t$ into a vector and retains only one of the r factor terms, hence eliminates the interaction effects between different factors. As we update the estimation of each individual loading vector $\boldsymbol{a}_{ik}$ separately in the algorithm, ideally this would remove the bias part in (20) which is due to the nonorthogonality of the loading vectors, and replace the eigengap λ_* in (20) by λ_i, as (15) only involves one eigenvector. It also leads to the elimination of the first two terms of $R^{(0)}$. As mentioned in Section 3, when updating $\widehat{\boldsymbol{a}}_{ik}^{(m)}$, we take advantages of the multiplicative nature of the project error $\xi_{ij}^{(m)}$ in (16), and the rapid growth of such benefits as the iteration number m grows. Thus we expect that the rate of HOPE estimators would become

$$\max_{1\le i\le r}\max_{1\le k\le K}\|\widehat{\boldsymbol{a}}_{ik}\widehat{\boldsymbol{a}}_{ik}^\top - \boldsymbol{a}_{ik}\boldsymbol{a}_{ik}^\top\|_{\mathrm{S}} \le C_{0,K}R^{(\mathrm{ideal})}, \tag{24}$$

where

$$R_{k,i}^{(\mathrm{ideal})} = \frac{\sigma^2}{\lambda_i}\sqrt{\frac{d_k}{T}} + \sqrt{\frac{\sigma^2 d_k}{\lambda_i T}}, \quad \text{and} \quad R^{(\mathrm{ideal})} = \max_{1\le i\le r}\max_{1\le k\le K} R_{k,i}^{(\mathrm{ideal})}. \tag{25}$$

Note that $R_{k,i}^{(\mathrm{ideal})}$ replaces all $d = d_1 \ldots d_K$ in the noise component of the stochastic error in (20) by d_k due to dimension reduction. The following theorem provides conditions under which this ideal rate is indeed achieved.

Let the statistical error bound of the initialization used in Algorithm 2 be ψ_0. For cPCA,

$$\psi_0 = \frac{\lambda_1\delta + R^{(0)}}{\lambda_*}, \tag{26}$$

where λ_* is the eigengap defined in (19) and $R^{(0)}$ is defined in (21).

Theorem 2 Suppose Assumptions 1, 2, 3 hold. Assume that $\delta_{\max} = \max_{k\le K}\delta_k < 1$ with δ_k defined in (9), and $r = O(T)$. Let $1/\gamma = 1/\gamma_1 + 2/\gamma_2$, $h \le T/4$, and $d = d_1\cdots d_K$. Suppose that for a proper numeric constant $C_{1,K}$ depending on K only, we have

$$\sqrt{1-\delta_{\max}} - (r^{1/2}+1)\psi_0/\sqrt{1-1/(4r)} > 0, \tag{27}$$

$$C_{1,K}\left(\frac{\lambda_1}{\lambda_r}\right)\psi_0^{2K-3} + C_{1,K}\sqrt{\frac{\lambda_1}{\lambda_r}}\left(\sqrt{\frac{r+\log T}{T}} + \frac{(r+\log T)^{1/\gamma}}{T}\right)\psi_0^{K-2} \le \rho < 1 \tag{28}$$

Then, after at most $M = O(\log\log(\psi_0/R^{(\text{ideal})}))$ iterations of Algorithm 2, in an event with probability at least $1-(Tr)^{-C}-\sum_k e^{-d_k}$, the HOPE estimator satisfies

$$\|\widehat{\boldsymbol{a}}_{ik}^{\text{iso}}\widehat{\boldsymbol{a}}_{ik}^{\text{iso}\top}-\boldsymbol{a}_{ik}\boldsymbol{a}_{ik}^{\top}\|_{\text{S}} \le C_{0,K}R^{(\text{ideal})}, \tag{29}$$

for all $1 \le i \le r, 1 \le k \le K$, where $C_{0,K}$ is a constant depending on K only and C is a positive numeric constant.

The detailed proof of the theorem is in online supplementary material, Appendix 1. The key idea of the analysis of HOPE is to show that the iterative estimator has an error contraction effect in each iteration. Theorem 2 implies that HOPE will achieve a faster statistical error rate than the typical $O_{\mathbb{P}}(T^{-1/2})$ whenever $\lambda_r \gg \sigma^2 \max_k d_k$. As $\text{vec}(\mathcal{X}_t)$ has d elements, the strong factors setting in the literature (R. Chen et al., 2022; Y. Han et al., 2020; Lam et al., 2011) typically assumes $\text{SNR} \asymp 1$. In our case it is similar to assuming the signal strength $\mathbb{E}\|\sum_{i=1}^r w_i f_{it} \otimes_{k=1}^K \boldsymbol{a}_{ik}\|_{\text{HS}}^2 \asymp \sigma^2 d$. When r is fixed and $\lambda_1 \asymp \cdots \asymp \lambda_r$, the statistical error rate will be reduced to $O_{\mathbb{P}}(T^{-1/2}d_{-k}^{-1/2})$, where $d_{-k} = \prod_{j\neq k} d_j$.

Remark 13 (Iteration complexity). Theorem 2 implies that Algorithm 2 achieves the desired estimation error $R^{(\text{ideal})}$ after at most $M = O(\log\log(\psi_0/R^{(\text{ideal})}))$ number of iterations. In this sense, after at most double-logarithmic number of iterations, the iterative estimator in Algorithm 2 converges to a neighbourhood of the true parameter $\boldsymbol{a}_{ik}$, up to a statistical error with a rate $O(R^{(\text{ideal})})$. We observe that Algorithm 2 typically converges within very few steps in practical implementations.

Remark 14 Condition (27) requires $r^{1/2}\psi_0$ to be small. It is a relatively strong condition due to the extra multiplier $r^{1/2}$ on the error of the initial estimators. This is a technical issue due to the need to invert the estimated $\boldsymbol{A}_k^{\top}\boldsymbol{A}_k$ in our analysis to construct the mode-k projection in Algorithm 2. In fact the $r^{1/2}$ term may be eliminated by applying a shrinkage procedure on the singular values of $\widehat{\boldsymbol{A}}_k$ after obtaining the updates of $\widehat{\boldsymbol{a}}_{ik}$, $1 \le i \le r$, similar to the procedure proposed by Anandkumar et al. (2014c).

Furthermore, the condition given by (28) originates from the multiplicative nature of the projection error $\zeta_{ij}^{(m)}$, as seen in (16) for $i \neq j$. If ψ_0 signifies the error bound for cPCA estimators, then condition (28) is satisfied when $(\lambda_1/\lambda_r)\psi_0^{2K-3} \lesssim 1$. In comparison, the iterative algorithm of Anandkumar et al. (2014c) requires that the initialization fulfills $\psi_0 \lesssim \lambda_r/\lambda_1 + 1/\sqrt{d_{\min}}$, a condition that is more stringent than (28). The ratio λ_1/λ_r in (28) is unavoidable. When updating the estimates of $\boldsymbol{a}_{ik}$ in Algorithm 2, we need to remove the effect of other factors ($j \neq i$) on the ith factor, which introduces the ratio of factor strengths λ_1/λ_r in the analysis.

In particular, if $\lambda_1 \asymp \cdots \asymp \lambda_r$, the shrinkage procedure can reduce conditions (27) and (28) to

$$C_{1,K}\psi_0 < 1, \tag{30}$$

where ψ_0 is the cPCA error bound in (26). It ensures that, with high probability, $\|\widehat{\boldsymbol{a}}_{ik}^{(0)}\widehat{\boldsymbol{a}}_{ik}^{(0)\top}-\boldsymbol{a}_{ik}\boldsymbol{a}_{ik}^{\top}\|_{\text{S}}$ are sufficiently small, so that the cPCA initialization is sufficiently close to the ground truth as in (30).

Remark 15 (Comparison with general tensor CP-decomposition methods). To estimate $\boldsymbol{a}_{ik}$ in (5), one can use the standard tensor CP-decomposition algorithms, such as those in Anandkumar et al. (2014c), Hao et al. (2020), and Sun et al. (2017), without utilizing the special features of TFM-cp. The randomized initialization estimators in these algorithms typically require the

incoherence condition $\vartheta_{\max} \lesssim \text{poly}\log(d_{\min})/\sqrt{d_{\min}}$. In contrast, the condition for ISO needs $\vartheta_{\max} \lesssim r^{-5/(2K)}$, which is weaker when $r = o(d_{\min}^{K/5})$. Similarly, we prove that the cPCA yields useful estimates when $r^2\vartheta_{\max}^K$ is small, or $\vartheta_{\max} \lesssim r^{-2/K}$. In other words, as long as r is not exceedingly large (e.g. $r = o(d_{\min}^{K/5})$), both cPCA and ISO permit a more lenient incoherence condition among the CP basis. Furthermore, the high-order coherence in TFM-cp leads to an impressive computational super-linear convergence rate of Algorithm 2, which is faster than the computational linear convergence rate of the iterative projection algorithm in Y. Han et al. (2020) or other variants of alternating least-squares approaches in the literature, that are at most linear with the required number of iterations $M = O(\log(\psi_0/R^{(\text{ideal})}))$.

Remark 16 (Comparison between TFM-cp and TFM-tucker Models). As discussed in Remark 3, TFM-cp can be written as a TFM-tucker with a special structure. One can ignore the special structure and treat it a generic TFM-tucker in (3) and estimate the loading spaces spanned by $\{\boldsymbol{a}_{ik}, 1 \le i \le r\}$ using the iterative estimation algorithm in Y. Han et al. (2020). In fact, ISO (as detailed in Algorithm 2) can be viewed as an enhancement of the iterative algorithm presented in Y. Han et al. (2020) to utilize the special structure of TFM-cp. This is achieved by permitting nonorthogonality in $\boldsymbol{A}_k$ and estimating each $\boldsymbol{a}_{ik}$, $1 \le i \le r$, individually.

Here we provide a brief comparison in the estimation accuracy between the estimators under these two settings to show the impact of the additional structure in TFM-cp. Note that for TFM-tucker, only the linear space spanned by $\boldsymbol{A}_k$ can be estimated hence the estimation accuracy is based on a specific space representation, different from that for the TFM-cp. For simplicity, we consider the case $\lambda_1 \asymp \cdots \asymp \lambda_r$.

(i) The iterative refinement algorithm (Algorithm 2) for TFM-cp requires similar conditions on the initial estimators as the iterative projection algorithms for TFM-tucker. Under many situations, both methods only require the initialization to retain a large portion of the signal, but not the consistency.

(ii) The statistical error rate of HOPE in (29) is the same as the upper bound of the iterative projection algorithms for estimation of the fixed rank TFM-tucker, c.f. Corollary 3.1 and 3.2 in Y. Han et al. (2020), which is shown to have the minimax optimality. It follows that HOPE also achieves the minimax rate-optimal estimation error under fixed r.

(iii) When the rank r diverges and SNR $\asymp 1$ where SNR is defined in (22), the estimation error of the loading spaces by the iterative estimation procedures, iTOPUP and TIPUP-iTOPUP procedures in Y. Han et al. (2020) applied to the specific TFM-tucker model implied by the TFM-cp model, is of the order $O_{\mathbb{P}}(\max_k r^{3K/2-1}T^{-1/2}d_{-k}^{-1/2})$, a rate that is always larger than $O_{\mathbb{P}}(\max_k r^{1/2}T^{-1/2}d_{-k}^{-1/2})$, the error rate of HOPE for TFM-cp model. The iTIPUP procedure for TFM-tucker model is $O_{\mathbb{P}}(\max_k r^{1/2+(K-1)\zeta}T^{-1/2}d_{-k}^{-1/2})$ where ζ controls the level of signal cancellation (see Y. Han et al., 2020 for details). When there is no signal cancellation, $\zeta = 0$, the rate of the two procedures are the same. Note that iTIPUP only estimates the loading space, while HOPE provides estimates of the unique loading vectors. The error rate of HOPE is better when $\zeta > 0$. This demonstrates that HOPE is able to utilize the specific structure in TFM-cp to achieve more accurate estimation than simply applying the estimation procedures designed for general TFM-tucker.

(iv) It can be seen that, computationally, the complexities for the initialization of both TFM-cp and TFM-tucker are the same, yet, the per iteration

complexity of TFM-cp is lower than that of TFM-tucker by a factor of r^{2K-2} where $r_1 = \ldots = r_K = r$ in TFM-tucker model.

Theorem 3 Suppose Assumptions 1, 2, 3 hold. Assume that $\delta_k < 1$ with δ_k defined in (9), $\sigma^2 \lesssim \lambda_r$ and condition (27) holds. Let $d_{\max} = \max_k d_k$. Then the HOPE estimator in Algorithm 2 using a specific h satisfies:

$$w_i^{-1}\left|\widehat{w}_i^{\text{iso}}\widehat{f}_{it}^{\text{iso}} - w_i f_{it}\right| = O_{\mathbb{P}}\left(\sqrt{\frac{\sigma^2}{\lambda_r}} + \sqrt{\frac{\sigma^2 d_{\max}}{\lambda_r T}}\right) \tag{31}$$

and

$$w_i^{-1}w_j^{-1}\left|\frac{1}{T-h_*}\sum_{t=h_*+1}^{T}\widehat{w}_i^{\text{iso}}\widehat{w}_j^{\text{iso}}\widehat{f}_{it-h_*}^{\text{iso}}\widehat{f}_{jt}^{\text{iso}} - \frac{1}{T-h_*}\sum_{t=h_*+1}^{T} w_i w_j f_{it-h_*} f_{jt}\right| = O_{\mathbb{P}}\left(\sqrt{\frac{\sigma^2 d_{\max}}{\lambda_r T}}\right) \tag{32}$$

for $1 \le i, j \le r$, $1 \le t \le T$ and all $1 < h_* \le T/4$.

Theorem 3 specifies the convergence rate for the estimated factors f_{it}. When $\lambda_r \gg \sigma^2 d_{\max} + T$, $w_i^{-1}|\widehat{w}_i^{\text{iso}}\widehat{f}_{it}^{\text{iso}} - w_i f_{it}|$ is much smaller than the parametric rate $T^{-1/2}$. If all the factors are strong (Lam et al., 2011) such that $\lambda_1 \asymp \lambda_r \asymp \sigma^2 d$, (31) implies that $w_i^{-1}|\widehat{w}_i^{\text{iso}}\widehat{f}_{it}^{\text{iso}} - w_i f_{it}| = O_{\mathbb{P}}(d^{-1/2} + d_{\max}^{1/2} d^{-1/2} T^{-1/2})$. Then, as long as $d_k \to \infty$ and $K \ge 2$, the estimated factors are consistent, even under a fixed T. In comparison, the convergence rate of the estimated factors in Theorem 1 of Bai (2003) for vector factor models is $O_{\mathbb{P}}(d^{-1/2} + T^{-1})$. Moreover, (32) shows that the error rates for the sample autocross-moment of the estimated factors to the true sample autocross-moment is also $o_{\mathbb{P}}(T^{-1/2})$ when $\lambda_r \gg \sigma^2 d_{\max}$. This implies that it is a valid option to use the estimated factor processes as the true factor processes to model the dynamics of the factors. When the estimation of these time series models only required autocorrelation and partial autocorrection functions, the results are expected to be the same as using the true factor process, without loss of efficiency. The statistical rates in Theorem 3 lay a foundation for further modelling of the estimated factor processes with vast repository of linear and nonlinear options.

5 Simulation studies

5.1 Alternative algorithms for estimation of TFM-cp

Here we present two alternative estimation algorithms for TFM-cp, by extending the popular rank one ALS algorithm of Anandkumar et al. (2014c) and orthogonalized alternating least square (OALS) of Sharan and Valiant (2017) designed for CP decomposition of noisy tensors, because $\widehat{\boldsymbol{\Sigma}}_h$ in (5) is indeed in a CP form, but with repeated components. In addition, we use cPCA estimates for initialization, instead of randomized initialization used for general CP decomposition. We will denote the algorithms as cALS (Algorithm 3) and cOALS (Algorithm 4), respectively. The simulation study below shows that, although cALS and cOALS perform better than the straightforward implementation of ALS and OALS with randomized initialization, they do not perform as well as the proposed HOPE algorithm. Hence we do not investigate their theoretical properties in this paper.

5.2 Simulation

In this section, we compare the empirical performance of different procedures of estimating the loading vectors of TFM-cp, under various simulation setups. We consider the cPCA initialization (Algorithm 1) alone, the iterative procedure HOPE, and the intermediate output from the iterative procedure when the number of iteration is 1 after initialization. The one-step procedure will be

Algorithm 3 cPCA-initialized Rank One Alternating Least Square (cALS)

Input: Observations $\mathcal{X}_t \in \mathbb{R}^{d_1 \times \cdots \times d_K}$ for $t = 1, ..., T$, the number of factors r, the time lag h, the cPCA initial estimate $(\widehat{\boldsymbol{a}}_{i1}^{\text{cpca}}, ..., \widehat{\boldsymbol{a}}_{iK}^{\text{cpca}})$, $1 \le i \le r$, the tolerance parameter $\epsilon > 0$, and the maximum number of iterations M.

1: Compute $\widehat{\boldsymbol{\Sigma}}_h$ defined in (6).

2: Initialize unit vectors $\widehat{\boldsymbol{a}}_{ik}^{(0)} = \widehat{\boldsymbol{a}}_{ik}^{\text{cpca}}$ for $1 \le k \le K$, $1 \le i \le r$. Set $m = 0$.

3: **for** $i = 1$ to r.

4: **repeat**

5: Set $m = m + 1$.

6: $k = 1$ to K.

7: Compute $\widetilde{\boldsymbol{a}}_{ik}^{(m)} = \widehat{\boldsymbol{\Sigma}}_h \times_{\ell=1}^{k-1} \widehat{\boldsymbol{a}}_{i\ell}^{(m)\top} \times_{\ell=k}^{K} \widehat{\boldsymbol{a}}_{i\ell}^{(m-1)\top} \times_{\ell=K+1}^{K+k-1} \widehat{\boldsymbol{a}}_{i\ell}^{(m)\top} \times_{\ell=K+k+1}^{2K} \widehat{\boldsymbol{a}}_{i\ell}^{(m-1)\top}$, where $\widehat{\boldsymbol{a}}_{i\ell}^{(m-1)} = \widehat{\boldsymbol{a}}_{i\ell-K}^{(m-1)}$ for $\ell > K$.

8: Compute $\widehat{\boldsymbol{a}}_{ik}^{(m)} = \widetilde{\boldsymbol{a}}_{ik}^{(m)} / \|\widetilde{\boldsymbol{a}}_{ik}^{(m)}\|_2$.

9: **end for**

10: **until** $m = M$ or $\max_k \|\widehat{\boldsymbol{a}}_{ik}^{(m)}\widehat{\boldsymbol{a}}_{ik}^{(m)\top} - \widehat{\boldsymbol{a}}_{ik}^{(m-1)}\widehat{\boldsymbol{a}}_{ik}^{(m-1)\top}\|_{\text{S}} \le \epsilon$.

11: Let $\widehat{\boldsymbol{a}}_{ik}^{\text{cALS}} = \widehat{\boldsymbol{a}}_{ik}^{(m)}$, $1 \le k \le K$.

12: **end for**

Output: $\widehat{\boldsymbol{a}}_{ik}^{\text{cALS}}$, $i = 1, ..., r$, $k = 1, ..., K$.

Algorithm 4 cPCA-initialized Orthogonalized Alternating Least Square (cOALS)

Input: Observations $\mathcal{X}_t \in \mathbb{R}^{d_1 \times \cdots \times d_K}$ for $t = 1, ..., T$, the number of factors r, the time lag h, the cPCA initial estimate $(\widehat{\boldsymbol{a}}_{i1}^{\text{cpca}}, ..., \widehat{\boldsymbol{a}}_{iK}^{\text{cpca}})$, $1 \le i \le r$, the tolerance parameter $\epsilon > 0$, the maximum number of iterations M.

1: Compute $\widehat{\boldsymbol{\Sigma}}_h$ defined in (6).

2: Initialize unit vectors $\widehat{\boldsymbol{a}}_{ik}^{(0)} = \widehat{\boldsymbol{a}}_{ik}^{\text{cpca}}$ for $1 \le k \le K$, $1 \le i \le r$. Set $\widehat{\boldsymbol{A}}_k^{(0)} = (\widehat{\boldsymbol{a}}_{1k}^{(0)}, ..., \widehat{\boldsymbol{a}}_{rk}^{(0)})$ and $m = 0$.

3: **repeat**

4: Set $m = m + 1$.

5: Find QR decomposition of $\widehat{\boldsymbol{A}}_k^{(m-1)}$, set $\widehat{\boldsymbol{A}}_k^{(m-1)} = \boldsymbol{Q}_k^{(m-1)}\boldsymbol{R}_k^{(m-1)}$ for $1 \le k \le K$.

6: **for** $k = 1$ to K.

7: Compute $\widehat{\boldsymbol{A}}_k^{(m)} = \text{mat}_k(\widehat{\boldsymbol{\Sigma}}_h)(*_{\ell \ne k}^{2K} \boldsymbol{Q}_\ell^{(m-1)})$, where $\boldsymbol{Q}_\ell^{(m-1)} = \boldsymbol{Q}_{\ell-K}^{(m-1)}$ for $\ell > K$ and * is the Khatri–Rao product.

8: **end for**

9: **until** $m = M$ or $\max_i \max_k \|\widehat{\boldsymbol{a}}_{ik}^{(m)}\widehat{\boldsymbol{a}}_{ik}^{(m)\top} - \widehat{\boldsymbol{a}}_{ik}^{(m-1)}\widehat{\boldsymbol{a}}_{ik}^{(m-1)\top}\|_{\text{S}} \le \epsilon$.

Output: $\widehat{\boldsymbol{a}}_{ik}^{\text{cOALS}} = \widehat{\boldsymbol{a}}_{ik}^{(m)}$, $i = 1, ..., r$, $k = 1, ..., K$.

denoted as 1HOPE. We also check the performance of the alternative algorithms ALS, OALS, cALS, and cOALS as described above. The estimation error shown is given by $\max_{i,k} \|\widehat{\boldsymbol{a}}_{ik}\widehat{\boldsymbol{a}}_{ik}^\top - \boldsymbol{a}_{ik}\boldsymbol{a}_{ik}^\top\|_{\text{S}}$.

We demonstrate the performance of all procedures under TFM-cp with $K = 2$ (matrix time series) with

$$\mathcal{X}_t = \sum_{i=1}^{r} w f_{it} \boldsymbol{a}_{i1} \otimes \boldsymbol{a}_{i2} + \mathcal{E}_t. \tag{33}$$

For $K = 2$ with model (33), we consider the following three experimental configurations:

(I) Set $r = 2$, $d_1 = d_2 = 40$, $T = 400$, $w = 6$ and vary δ in the set $[0, 0.5]$. The purpose of this setting is to verify the theoretical bounds of cPCA and HOPE in terms of the coherence parameter δ.

(II) Set $r = 2$, $d_1 = d_2 = 40$, $\delta = 0.2$. We vary the sample size T and the signal strength w to investigate the impact of δ against signal strength and sample size.
(III) Set $r = 3$, $d_1 = d_2 = 40$, $T = 400$, $w = 8$ and vary δ to check the sensitivities of δ for all the proposed algorithms and compare with randomized initialization.

Results from an additional simulation settings under $K = 2$ and $K = 3$ cases are given in online supplementary material, Appendix 2. We repeat all the experiments 100 times. For simplicity, we set $h = 1$.

The loading vectors are generated as follows. First, the elements of matrices $\widetilde{\boldsymbol{A}}_k = (\widetilde{\boldsymbol{a}}_{1k}, \ldots, \widetilde{\boldsymbol{a}}_{rk}) \in \mathbb{R}^{d_k \times r}$, $1 \le k \le K$, are generated from i.i.d. $N(0, 1)$ and then orthonormalized through QR decomposition. Then if $\delta = 0$, set $\boldsymbol{A}_k = \widetilde{\boldsymbol{A}}_k$, otherwise, set $\boldsymbol{a}_{1k} = \widetilde{\boldsymbol{a}}_{1k}$ and $\boldsymbol{a}_{ik} = (\widetilde{\boldsymbol{a}}_{1k} + \theta\widetilde{\boldsymbol{a}}_{ik})/\|\widetilde{\boldsymbol{a}}_{1k} + \theta\widetilde{\boldsymbol{a}}_{ik}\|_2$ for all $i \ge 2$ and $1 \le k \le K$, with $\vartheta = \delta/(r-1)$ and $\theta = (\vartheta^{-2/K} - 1)^{1/2}$. The commonly used incoherence measure (Anandkumar et al., 2014c; Hao et al., 2020) under this construction is $\vartheta_{\max} = (1 + \theta^2)^{-1/2} = \vartheta^{1/K}$.

The noise $\mathcal{E}_t$ in the model is white $\mathcal{E}_t \perp \mathcal{E}_{t+h}$, $h > 0$, and generated according to $\mathcal{E}_t = \Psi_1^{1/2} Z_t \Psi_2^{1/2}$ where all of the elements in the $d_1 \times d_2$ matrix Z_t are i.i.d. $N(0, 1)$. Furthermore, Ψ_1, Ψ_2 are the covariance matrices along each mode with the diagonal elements being 1 and all the off-diagonal elements being ψ_1, ψ_2. Throughout this section, we set the off-diagonal entries of the covariance matrices of the noise as $\psi_1 = \psi_2$.

Under Configurations I and II with $r = 2$, the factor processes f_{1t} and f_{2t} are generated as two independent AR(1) processes, following $f_{1t} = 0.8 f_{1t-1} + e_{1t}$, $f_{2t} = 0.6 f_{2t-1} + e_{2t}$. Under Configuration III and Configurations IV and V in online supplementary material, Appendix 2, with $r = 3$, f_{1t}, f_{2t}, f_{3t} are generated as independent AR(1) processes, with $f_{1t} = 0.8 f_{1t-1} + e_{1t}$, $f_{2t} = 0.7 f_{2t-1} + e_{2t}$, $f_{3t} = 0.6 f_{3t-1} + e_{3t}$. Here, all of the innovations follow i.i.d. $N(0, 1)$. The factors are not normalized.

Figure 1 shows the boxplots of the estimation errors for cPCA and HOPE under configuration I, for different δ. It can be seen that the performance of cPCA deteriorates as δ increases, while that of HOPE remains almost unchanged. The median of the cPCA estimation errors increases almost linearly with δ, with a R^2 of 0.977. This linear effect of δ on the performance bounds of cPCA is confirmed by the theoretical results in (20).

The experiment of Configuration II is conducted to verify the theoretical bounds on different sample sizes T and signal strengths w. Figures 2 and 3 show the logarithm of the estimation errors under different (w, T) combinations. It can be seen from Figure 2 that the estimation error of cPCA decreases to a lower bound as w and T increases. The lower bound is associated with the bias term in (20) that cannot be reduced by a larger w and T. This is the baseline error due to the nonorthogonality. In contrast, the phenomenon of HOPE is very different. Figure 3 shows that the performance improves monotonically as w or T increases. Again, this is consistent with the theoretical bounds in (29).

Figure 4 shows the boxplots of the logarithm of the estimation errors for 7 different methods with choices of δ under configuration III. ALS and OALS are implemented with $L = 200$ random initiations. It can be seen that HOPE outperforms all the other methods. Again, the choice of δ does not affect the performance of HOPE significantly. One-step method (1HOPE) is better than the cPCA alone, and the iterative method HOPE is in turn better than the one-step method. When the coherence δ decreases, all methods perform better, but the advantage of HOPE over one-step method and the advantage of one-step method over the cPCA initialization become smaller. For the extremely small $\delta = 0.01$, all loading vectors are almost orthogonal to each other. In this case, all the iterative procedures, including the one-step HOPE, perform similarly. In addition, ALS and cALS are always the worst under the cases $\delta \ge 0.1$. The hybrid methods cALS and cOALS improve the original randomized initialized ALS and OALS significantly, showing the advantages of the cPCA initialization. It is worth noting that cOALS has comparable performance with 1HOPE and HOPE when δ is small.

6 Applications

In this section, we demonstrate the use of TFM-cp model using the taxi traffic data set used in R. Chen et al. (2022). The data set was collected by the Taxi & Limousine Commission of

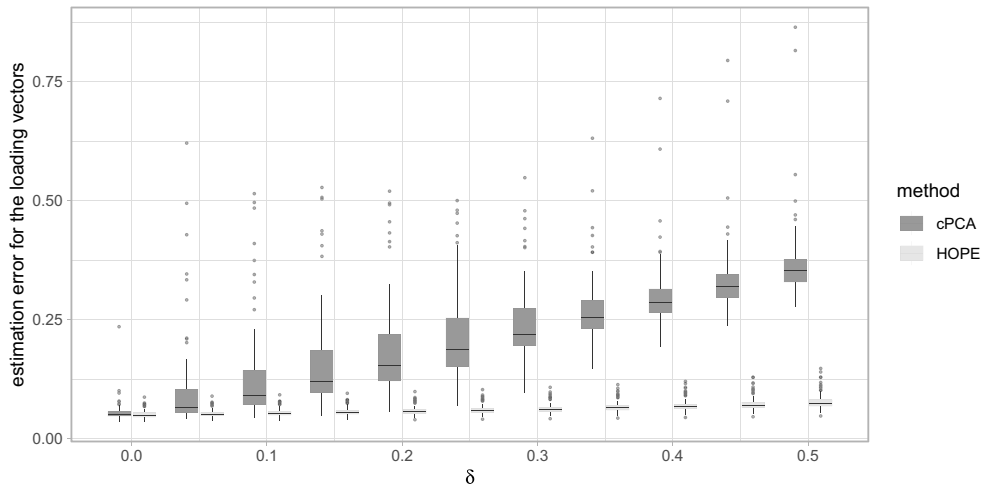

Figure 1. Boxplots of the estimation error over 100 replications under experiment configuration I with different δ.

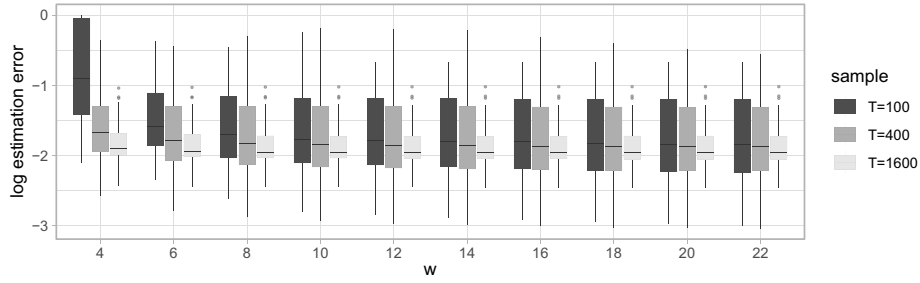

Figure 2. Boxplots of the logarithm of the estimation error for cPCA under experiment configuration II.

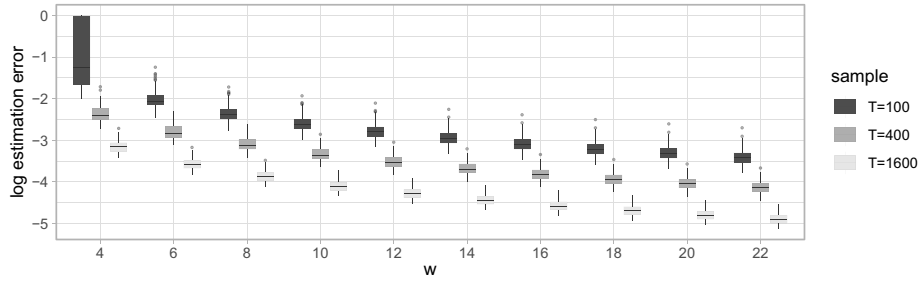

Figure 3. Boxplots of the logarithm of the estimation error for HOPE under experiment configuration II.

New York City, and published at https://www.nyc.gov/site/tlc/about/tlc-trip-record-data.page. Within Manhattan Island, it contains 69 predefined pick-up and drop-off zones and 24 hourly period for each day from 1 January 2009 to 31 December 2017. The total number of rides moving among the zones within each hour is recorded, yielding a $\mathcal{X}_t \in \mathbb{R}^{69\times69\times24}$ tensor for each day, using the hour of day as the third dimension.

One natural way to model the hourly traffic data is to consider $\boldsymbol{X}_t \in \mathbb{R}^{69\times69}$ for $t = 1, \ldots, 24 \times N$, where N is the total number of days. It is apparent that such hourly data have two types of seasonality: weekly seasonality and daily seasonality. We handle the weekly

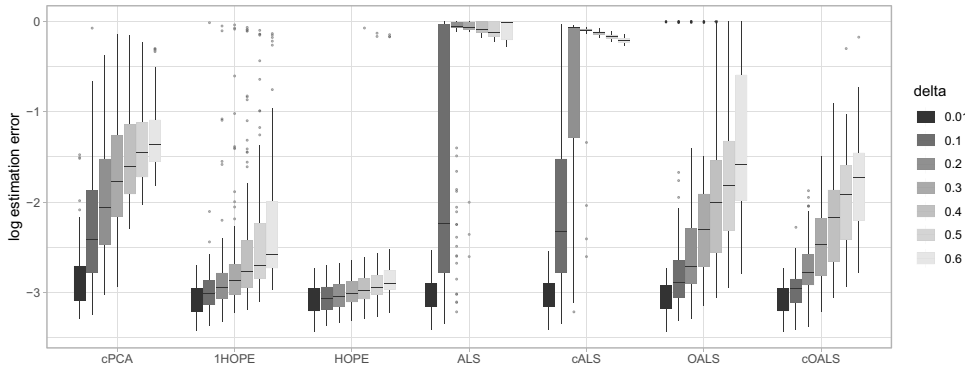

Figure 4. Boxplots of the logarithm of the estimation error under experiment configuration III. Seven methods with seven choices of δ are considered in total.

seasonality by separating the time series into two parts: business-day series and nonbusiness-day series. The length of the business-day series is 2,262 days, and that of the nonbusiness-day is 1,025 days. The daily seasonality is a more interesting and important issue. It is clear that taxi usage heavily depends on time of the day (morning and evening rush hours, lunch hours, etc). Seasonal time series models have been extensively studied for univariate time series (Box & Jenkins, 1976; Reinsel, 2003; Shumway & Stoffer, 2006; Tsay, 2005), but these parametric models are difficult to be extended to deal with high dimensional matrix time series. Segmenting the 24-hour day into distinct intervals, such as morning rush hours and business hours, loses the detailed hourly information and requires pre-determined segmentation scheme (Zhang & Wang, 2019; Zhu et al., 2022). Here we adopt the nonparametric approach by stacking the hourly observations into a (multidimensional) daily observation. This is equivalent to turning hourly observations into daily 24-dimensional vector observations in the univariate time series case, a commonly used approach. By jointly modelling the 24-hourly observations within the day together, the detailed daily pattern can be captured more accurately and more flexibly without a parametric model assumption. This way, the daily pattern is built into the model simultaneously, and the interaction among the geographic and temporal patterns will be revealed.

After some exploratory analysis, we decide to use the TFM-cp with $r = 4$ factors for both business-day series and nonbusiness-day series, and estimate the model with $h = 1$. For the nonbusiness-day series, TFM-cp explains 63.0% of the variability in the data. In comparison, treating the tensor time series as a 114,264 ($=69 \times 69 \times 24$) dimensional vector time series, the traditional vector factor model with 4 factors explains about 90.0% variability, but uses $4 \times 114,264$ parameters for the loading matrix. R. Chen et al. (2022) used TFM-tucker with $4 \times 4 \times 4$ core factor tensor process. Using iTIPUP estimator of Y. Han et al. (2020), TFM-tucker explains 80.1% variability. Similarly, for the business-day series, the explained fractions of variability by the TFM-cp with 4 factors, vector factor model with 4 factors, and TFM-tucker with $4 \times 4 \times 4$ core factor tensor are 68.1%, 90.9%, 84.0%, respectively. Comparison of model complexity between TFM-cp and TFM-tucker is substantially more complex compared to that of traditional tensor decomposition, owing to the stochastic nature of the latent factor process. Since both models are estimated through the sample autocovariance tensor, we may count, under each model, the number of parameters required in the population version of the autocovariance tensor. Both models require $(69 + 69 + 24) \times 4$ number of parameters in terms of loading matrices or vectors, though the degree of freedom for TFM-tucker is slightly smaller due to orthonormal requirements of the loading matrices. However, for the lag-1 autocovariance of the factor processes, TFM-cp only requires four parameters (for the four uncorrelated factor process) while TFM-tucker requires $(4 \times 4 \times 4)^2$ parameters (minus certain savings from rotation ambiguity). More importantly, TFM-cp harbours a much smaller dimensional factor process ($\boldsymbol{f}_t \in \mathbb{R}^4$) than TFM-tucker ($\mathcal{F}_t \in \mathbb{R}^{4\times4\times4}$), thereby simplifying subsequent modelling of the latent factor process.

The literature on Markov process models, for example Zhang and Wang (2019) and Zhu et al. (2022), regards each trip as a transition from the pickup location to the drop-off location,

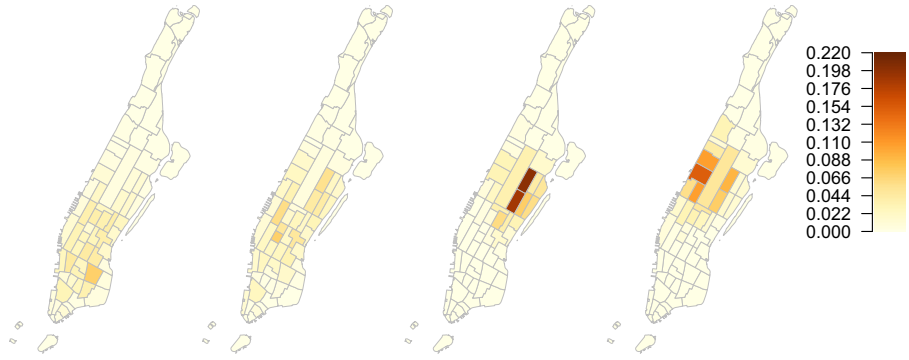

Figure 5. Loadings on four pickup factors for business-day series.

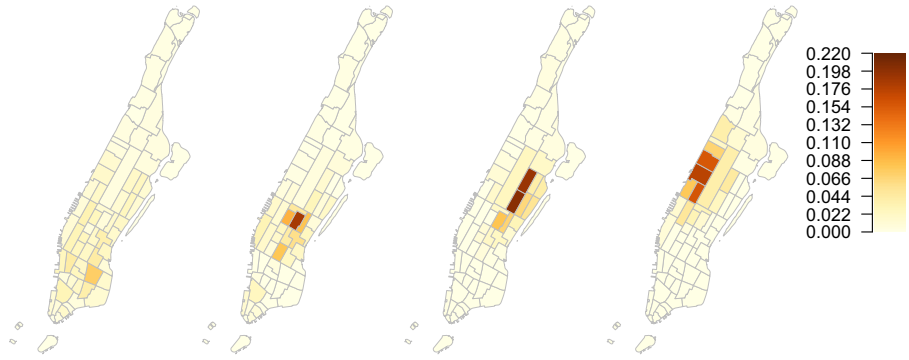

Figure 6. Loadings on four drop-off factors for business-day series.

rendering the data as a collection of fragmented sample paths representative of a city-wide Markov process. When one aggregates the individual trips within a time period to form a traffic volume matrix, the model, with an assumed fixed (reduced rank) Markov transition matrix, essentially induces an order-1 autoregressive model on the volume matrix time series. Therefore, the distinction between the Markov process models and the CP factor models parallels that between autoregressive models and factor models.

Figures 5 and 6 show the heatmap of the estimated loading vectors $(\boldsymbol{a}_{11}, \ldots, \boldsymbol{a}_{41})$ (related to pick-up locations) and $(\boldsymbol{a}_{12}, \ldots, \boldsymbol{a}_{42})$ (related to drop-off locations) of the 69 zones in Manhattan, respectively, for the business-day series. Table 1 shows the corresponding loading vectors $(\boldsymbol{a}_{13}, \ldots, \boldsymbol{a}_{43})$ on the time of day dimension. For a more meaningful interpretation, we have re-scaled the loading vectors $\boldsymbol{a}_{ik}$ and the factors $w_i f_{it}$ such that $\|\boldsymbol{a}_{ik}\|_1 = 1$, for $1 \le i \le 4$, $1 \le k \le 3$. Figure 7 shows the estimated four factors $(w_i f_{it})$ for business-day series in 1,000. (Please note the significant difference in scale.) For a more detailed examination, we show the four factor series in the third year (year 2011) in online supplementary material, Figure 3 in Appendix 3.

It is seen that the estimated loading vectors and the factors are predominantly positive, although there are a few small negative values which we will ignore. When the loading vectors are scaled to sum to 1 (hence percentages), the model has the following interesting interpretation. First, the expected daily total volume $(\sum_{i,j,k} X_{t,ijk})$ is the sum of the four factors $w_1 f_{1t} + \ldots + w_4 f_{4t}$. Hence the daily traffic volumes essentially consist of taxi rides following four different patterns, each corresponding to the rank-1 tensor $\boldsymbol{a}_{i1} \otimes \boldsymbol{a}_{i2} \otimes \boldsymbol{a}_{i3}$, $i = 1, \ldots, 4$. One may also imagine that there are four types of taxi users in the city, each following one specific traffic pattern (of course an individual may take multiple trips in a day and follow different patterns for each trip).

Table 1. Estimated four loading vectors $\boldsymbol{a}_{i3} \in \mathbb{R}^{24}$ ($i = 1, \ldots, 4$), for hour of day mode

0 to 24		2		4		6		8		10		12		2		4		6		8		10	12
$i = 1$	5	3	2	1	1	1	1	2	2	3	3	3	3	4	4	4	3	5	7	8	9	9	9	8
$i = 2$	0	0	0	0	1	3	12	16	13	11	7	6	5	4	4	3	2	2	3	3	2	2	1	1
$i = 3$	1	0	0	0	0	0	2	5	7	6	6	7	7	7	7	8	6	7	7	6	4	3	2	1
$i = 4$	1	1	0	0	0	0	1	3	5	4	5	5	6	6	7	7	6	8	9	8	6	5	4	3

Note. Business day. Values are in percentage.

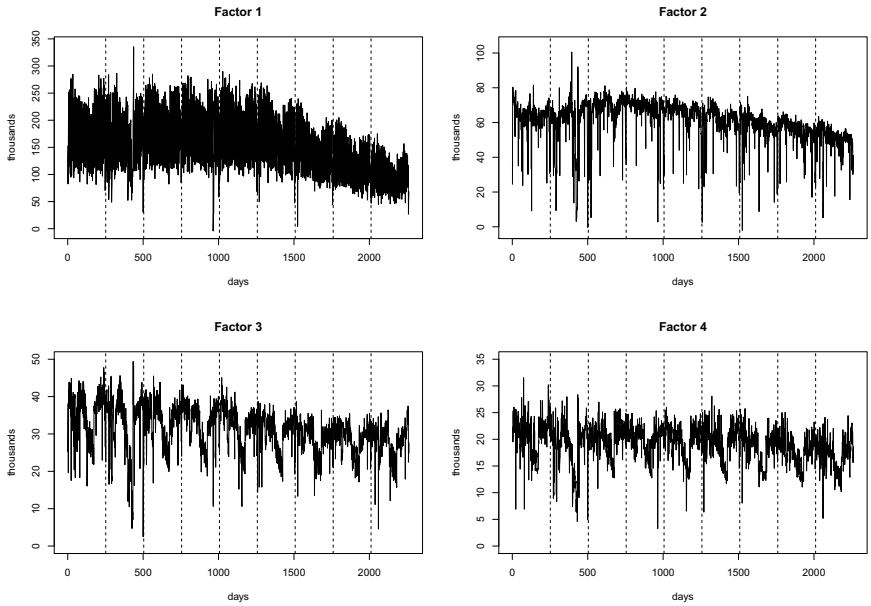

Figure 7. Estimated four factors for business-day series.

It is interesting to study the component of the rank-1 tensor $\boldsymbol{a}_{i1} \otimes \boldsymbol{a}_{i2} \otimes \boldsymbol{a}_{i3}$. Specifically, $\boldsymbol{a}_{i3}$ shows how the total volume of traffic pattern i ($w_i f_{it}$) is distributed to different hours of the day. For example, Table 1 shows that 16% of pattern 2 volume $w_2 f_{2t}$ is allotted to between 7a.m. and 8a.m., while only 3% of pattern 4 volume $w_4 f_{4t}$ is allotted to that hour. The rank-1 matrix $\boldsymbol{a}_{i1}\boldsymbol{a}_{i2}^{\top}$ shows the spatial pattern of traffic pattern i, where $\boldsymbol{a}_{i1}$ shows the percentage of pattern i volume ($w_i f_{it}$) at each hour being picked up in each location, and $\boldsymbol{a}_{i2}$ shows the percentage of pattern i volume $w_i f_{it}$ being dropped-off in each location, and $a_{i1k}a_{i2\ell}$ is the percentage of pattern i volume $w_i f_{it}$ from location k to location ℓ. This spatial pattern does not change through the day, but the volume in each hour is controlled by $\boldsymbol{a}_{i3} w_i f_{it}$.

From Table 1, it is seen that Factor 1 (or traffic pattern 1) roughly corresponds to the evening hours of 6p.m. to 12a.m., by the loading vector $\boldsymbol{a}_{13}$, with main activities in the SoHu and lower east side as both the pick-up and drop-off locations. From the estimated factor series plot in Figure 7, it seems that this traffic pattern (pattern 1) has the largest overall volume, but with a very strong yearly seasonal pattern and a large daily variation. Intuitively, people use less taxi service when the weather is nice, hence the volume is relatively small in summer and early fall, even though there are more evening activities in the summer. The large daily variation is

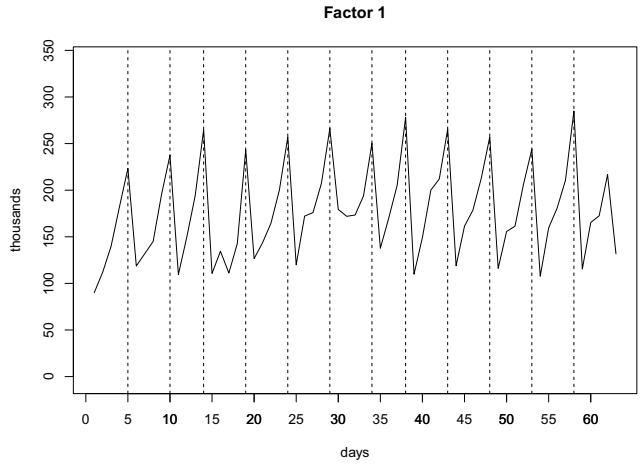

Figure 8. Estimated Factor 1 for business-day series in the 3-months business-day period from 1 January 2011 to 31 March 2011. Vertical lines mark the end of business week.

due to a weekly effect. Figure 8 shows the 3-month business-day period from 1 January 2011 to 31 March 2011, in which the vertical line marks the end of working week (Friday or the day before holiday). It is clearly seen that, for this mainly evening-activity traffic pattern, the volume in the end of working week is almost twice as large as that in the beginning of the working week.

Again from Table 1, it is seen that Factor 2 (or traffic pattern 2) roughly corresponds to the morning rush hours of 6a.m. to 12a.m., by the loading vector $\boldsymbol{a}_{23}$, with main activities in the midtown area as the pick-up locations, and Times square and 5th Avenue as the drop-off locations. About 23.1% (defined as $\sum_t w_2 f_{2t} / \sum_t \sum_i w_i f_{it}$) of the total traffic follows this pattern. From Figure 7, it is seen that Factor 2 is quite stable throughout the year, which is again intuitively understandable as the traffic pattern is mainly used by the steady population of people commuting to work in morning rush hours. There is a large number of (small value) outliers, most of them corresponding to the business days before or after major holidays. It can be seen more clearly from online supplementary material, Figure 3 in Appendix 3.

For Factors 3 and 4, the areas that load heavily on the factors for pick-up are quite similar to that for drop-off, i.e. upper east side (with affluent neighbourhoods and museums) on Factor 3, and upper west side (with affluent neighbourhoods and performing arts) on Factor 4. The conventional business hours are heavily and almost exclusively loaded on these factors. From Figure 7, it seems that both patterns have a yearly seasonal effect, small in the summer and early fall, which can be seen more clearly in online supplementary material, Figure 3 in Appendix 3. Their volumes are relatively small than that of Factors 1 and 2.

We note that TFM-cp representation is unique which facilitates a more 'unique' interpretation. On the other hand, TFM-tucker is subject to arbitrary rotation. Using TFM-tucker to analyse the same data set, R. Chen et al. (2022) used varimax rotation to obtain one specific representation of their estimated model and provided interesting interpretations. Their results are quite different from that of TFM-cp. First, since TFM-tucker representation requires orthonormal loading matrices, the discovered patterns in the loading matrices are forced to be different. For example, the daily patterns revealed in R. Chen et al. (2022) have quite distinct periods, while Table 1 shows more intertwined (nonorthogonal) patterns. Second, TFM-tucker requires $4 \times 4 \times 4$ factor processes. The column loading vectors in each loading matrices work on all these factors, instead of on only one factor as in TFM-cp. The interpretation of these loading vectors are more convoluted. For example, in TFM-tucker, the volume from all four heavily loaded pick-up areas identified by $\boldsymbol{a}_{i1}$, $i = 1, \ldots, 4$ can be travelling to all four heavily loaded drop-off areas $\boldsymbol{a}_{i2}$, $i = 1, \ldots, 4$. But in TFM-cp, the rank-1 matrix $\boldsymbol{a}_{i1}\boldsymbol{a}_{i2}^{\top}$ shows the exact proportion of pattern i traffic from each of the pick-up area identified by $\boldsymbol{a}_{i1}$ to the drop-off area $\boldsymbol{a}_{i2}$. In particular, it is seen that $\boldsymbol{a}_{i1}$ is very similar to $\boldsymbol{a}_{i2}$ for $i = 1, 3, 4$. This observation suggests that our TFM-cp model may offer

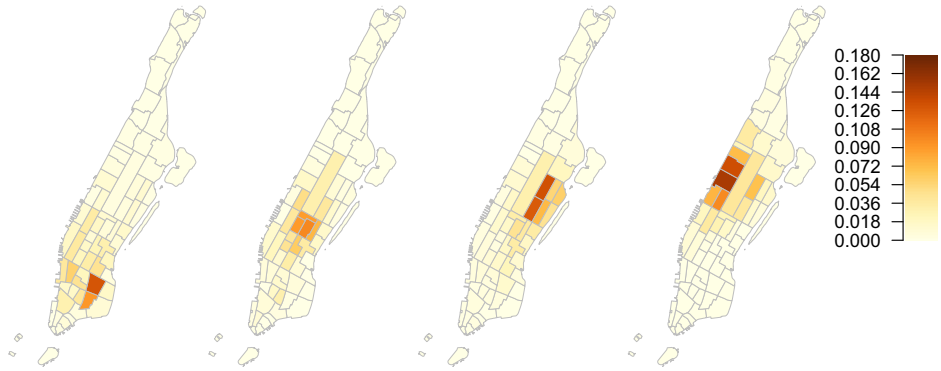

Figure 9. Loadings on four pick-up factors for nonbusiness-day series.

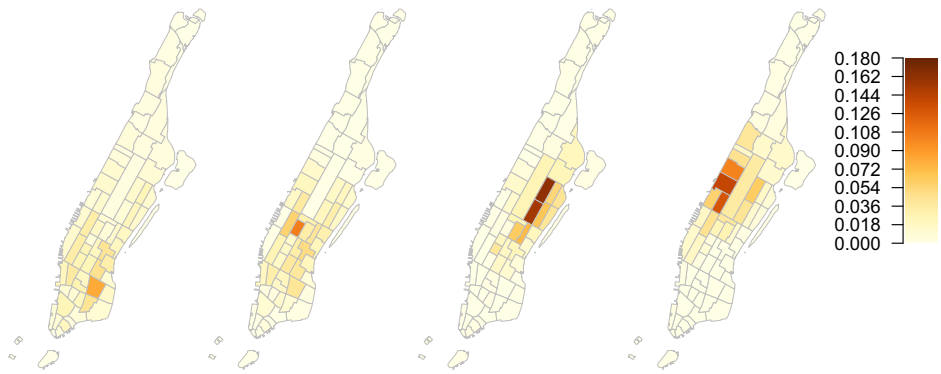

Figure 10. Loadings on four drop-off factors for nonbusiness-day series.

better intuitive understanding as it aligns with the expectation that most taxi traffic activities are likely confined within specific areas. This comparison further underscores the distinct analytical insights offered by the TFM-cp model in capturing the spatial–temporal dynamics of urban taxi traffic.

For the nonbusiness-day series, the estimated loading vectors $(\boldsymbol{a}_{11}, \ldots, \boldsymbol{a}_{41})$ (related to pick-up locations), $(\boldsymbol{a}_{12}, \ldots, \boldsymbol{a}_{42})$ (related to drop-off locations), $(\boldsymbol{a}_{13}, \ldots, \boldsymbol{a}_{43})$ (on the hour of day dimension), and the estimated factors $(w_1 f_{1t}, \ldots, w_4 f_{4t})$ are showed in Figures 9, 10, Table 2, and Figure 11, respectively. Understandably, the morning rush hour pattern in the business-day series (Factor 2) disappears here but the night-time pattern (Factor 1) now lasts deep into the early hours, comparing Tables 1 and 2. From Figure 11, it is seen that there exist two different yearly seasonal patterns. Factors 3 and 4 are similar to that of business-day series, with small volumes in the summer and fall, again confirming that the use of taxi service is relatively low when the weather is good for walking in the city. On the other hand, the volume of Factor 2 is typically small in the winter time. The volumes of night-life pattern in Factor 1 remain to be volatile. It has many small-value outliers, mostly on the day before a business day (Sundays or the end of holiday.) These can be seen more clearly in the more detailed Figure 12, which shows the estimated factors of all the nonbusiness days in Year 2011 (year 3), with vertical lines indicating the day before a business day (dashed lines for Sundays and solid lines for Mondays of long weekend when Tuesday is the start of business week.) This is again intuitively understandable, because people tend not to stay out too late if they need to work the next day.

The pick-up and drop-off locations that heavily load on Factors 1, 3, 4 are similar to that for Factors 1, 3, 4 in the business-day series. The daytime hours load on Factors 3 and 4, and the

Table 2. Estimated four loading vectors $\boldsymbol{a}_{i3} \in \mathbb{R}^{24}$ ($i = 1, \ldots, 4$) for hour of day mode

0 to 24		2		4		6		8		10		12		2		4		6		8		10	12
$i = 1$	10	10	9	7	3	1	1	1	1	2	2	3	3	3	3	3	3	4	5	5	5	5	5	6
$i = 2$	4	3	2	1	1	1	1	1	2	2	3	4	6	8	7	6	5	6	8	10	6	5	5	4
$i = 3$	2	1	1	1	0	0	1	2	3	6	7	8	8	8	8	8	7	6	6	5	4	3	2	2
$i = 4$	3	2	1	1	1	0	1	1	3	5	6	7	7	7	7	7	6	6	7	7	5	4	4	3

Note. Nonbusiness day. Values are in percentage.

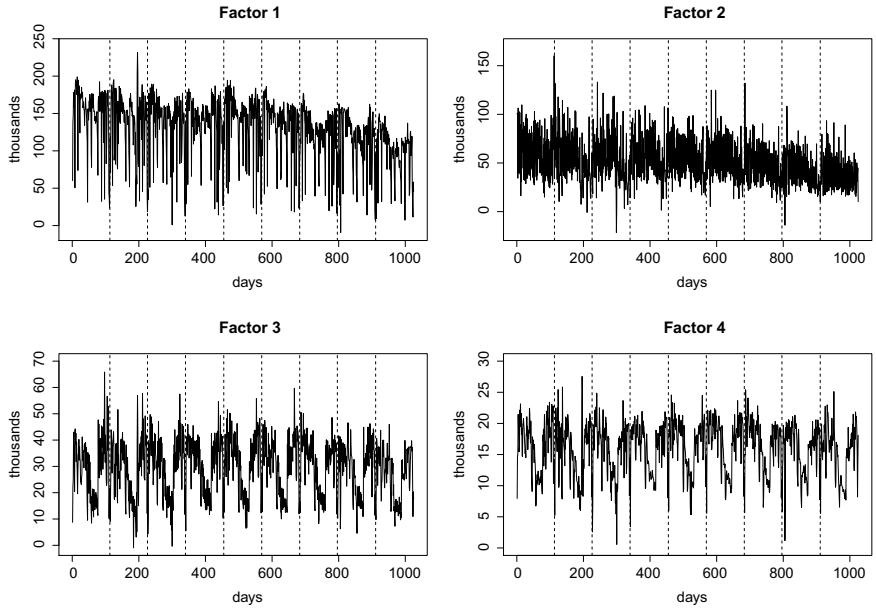

Figure 11. Estimated four factors for nonbusiness-day series.

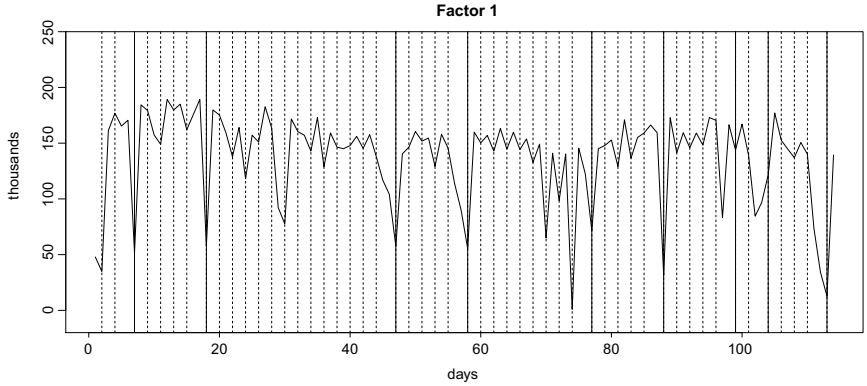

Figure 12. Estimated Factor 1 for nonbusiness-day series in Year 2011 with vertical lines indicating the day before a business day (dashed lines for Sundays and solid lines for Mondays of a long weekend.

night life hours from 12a.m. to 4a.m. load on Factor 1. As for the second factor, it loads heavily on midtown area for pick-up, on the lower west side near Chelsea (with many restaurants and bars) for drop-off, on the afternoon/evening hours between 1p.m. to 8p.m. as the dominating periods.

We remark that this example is just for illustration and showcasing the interpretation of the proposed tensor factor model. Again we note that for the TFM-tucker model, one needs to identify a proper representation of the loading space in order to interpret the model. In R. Chen et al. (2022), varimax rotation was used to find the most sparse loading matrix representation to model interpretation. For TFM-cp, the model is unique hence interpretation can be made directly. Interpretation is impossible for the vector factor model in such a high-dimensional case.

7 Discussion

In this paper, we propose a tensor factor model with a low-rank CP structure and develop its corresponding estimation procedures. The estimation procedure takes advantage of the special structure of the model, resulting in faster convergence rate and more accurate estimations comparing to the standard procedures designed for the more general TFM-tucker, and the more general tensor CP decomposition. Numerical study illustrates the finite sample properties of the proposed estimators. The results show that HOPE uniformly outperforms the other methods, when the observations follow the specified TFM-cp.

The HOPE in this paper is based on CP decomposition of the second moment tensor $\boldsymbol{\Sigma}_h = \sum_{i=1}^r \lambda_i (\otimes_{k=1}^K \boldsymbol{a}_{ik})^{\otimes 2}$, an order $2K$ tensor. The intuition that higher order tensors tend to have smaller coherence among the CP components leads to the consideration of using higher order cross-moments to have more orthogonal CP components. For example, let the mth cross moment tensor with lags $0 = h_1 < \cdots < h_m$ be

$$\boldsymbol{\Sigma}^{(m)}_{h_1 \cdots h_m} = \mathbb{E}\Big[\otimes_{j=1}^m \mathcal{X}_{t-h_j}\Big].$$

When the factor processes $f_{it}, i = 1, \ldots, r$ are independent across different i in TFM-cp, a naive 4th cross moment tensor to estimate $\boldsymbol{a}_{ik}$ is

$$\begin{aligned}\boldsymbol{\Sigma}^{(4)}_{h_1h_2h_3h_4} - \boldsymbol{\Sigma}^{(2)}_{h_1h_2} \otimes \boldsymbol{\Sigma}^{(2)}_{h_3h_4} - \mathbb{E}[\mathcal{X}_{t-h_1} \otimes \mathcal{X}^*_{t-h_2} \otimes \mathcal{X}_{t-h_3} \otimes \mathcal{X}^*_{t-h_4}] - \mathbb{E}[\mathcal{X}_{t-h_1} \otimes \mathcal{X}^*_{t-h_2} \otimes \mathcal{X}^*_{t-h_3} \otimes \mathcal{X}_{t-h_4}] \\ = \sum_{i=1}^r \lambda^{(4)}_{i,h_1h_2h_3h_4} (\otimes_{k=1}^K \boldsymbol{a}_{ik})^{\otimes 4},\end{aligned}$$

with $\{\mathcal{X}^*_t\}$ being an independent coupled process of $\{\mathcal{X}_t\}$ and when $h_j = (j-1)h$,

$$\lambda^{(4)}_{i,h_1h_2h_3h_4} = \mathbb{E}\prod_{j=0}^3 f_{i,t-jh} - [\mathbb{E}f_{i,t}f_{i,t-h}]^2 - [\mathbb{E}f_{i,t}f_{i,t-2h}]^2 - [\mathbb{E}f_{i,t}f_{i,t-h}][\mathbb{E}f_{i,t}f_{i,t-3h}].$$

This naive 4th cross moment tensor has more orthogonal CP bases. In light tailed case, simulation shows that it is much worse than the second moment tensor, due to the reduced signal strength $\lambda^{(4)}_{i,h_1h_2h_3h_4}$. However, for heavy tailed and skewed data, this procedure would be helpful. It would be an interesting and challenging problem to develop an efficient higher cross moment tensor to improve the statistical and computational performance. We leave this for future research.

Our primary consideration was directed towards the CP factor model in a time series setting, as the need of effectively analysing tensor time series has arisen in many applications, and CP factor model is an efficient approach for such analysis. Without a specified (parametric) model for the latent factor processes (an topic currently under investigation), we focus on the autocovariance and autocross-moment tensor for effective estimation of the proposed model. This is the main contribution of the paper. However, the proposed cPCA and ISO can be used directly in or be extended to many other problems involving CP decomposition of certain

type of tensors. For example, in many problems where higher order moments can be introduced to reduce incoherence, the cPCA and ISO algorithms may offer better initialization and outperform conventional tensor power iteration methods. One specific example is the kurtosis tensor in independent component analysis in Auddy and Yuan (2023a). Another possible extension is highlighted in Remark 12 where we pointed out how cPCA can be modified to deal with situations when a few leading singular values of the autocross-moment tensor are the same.

Acknowledgments

We would like to express our sincere gratitude to the referees and the editors for their invaluable feedback, which greatly enhanced the quality of this paper.

Conflict of interests: None declared.

Funding

Y.H.'s research was supported in part by US National Science Foundation grant IIS-1741390. D.Y.'s research was supported in part by US National Science Foundation grant IIS-1741390, Hong Kong Research Grants Council grant GRF 17301620, Hong Kong Research Grants Council grant CRF C7162-20GF and Shenzhen City grant SZRI2023-TBRF-03. C.-H.Z.'s research was supported in part by US National Science Foundation grants DMS-1721495, IIS-1741390, CCF-1934924, DMS-2052949 and DMS-2210850. R.C.'s research was supported in part by US National Science Foundation grants DMS-1737857, IIS-1741390, CCF-1934924, DMS-2027855, and DMS-2319260.

Data availability

The taxi traffic data are available at https://www.nyc.gov/site/tlc/about/tlc-trip-record-data.page.

Supplementary material

Supplementary material is available online at *Journal of the Royal Statistical Society: Series B*.

References

Ahn S. C., & Horenstein A. R. (2013). Eigenvalue ratio test for the number of factors. *Econometrica*, *81*(3), 1203–1227. https://doi.org/10.3982/ECTA8968

Allman E. S., Matias C., & Rhodes J. A. (2009). Identifiability of parameters in latent structure models with many observed variables. *The Annals of Statistics*, *37*(6A), 3099–3132. https://doi.org/10.1214/09-AOS689

Alter O., & Golub G. H. (2005). Reconstructing the pathways of a cellular system from genome-scale signals by using matrix and tensor computations. *Proceedings of the National Academy of Sciences*, *102*(49), 17559–-17564. https://doi.org/10.1073/pnas.0509033102

Anandkumar A., Ge R., Hsu D., & Kakade S. M. (2014a). A tensor approach to learning mixed membership community models. *Journal of Machine Learning Research*, *15*(1), 2239–2312. https://jmlr.csail.mit.edu/papers/v15/anandkumar14a.html

Anandkumar A., Ge R., Hsu D., Kakade S. M., & Telgarsky M. (2014b). Tensor decompositions for learning latent variable models. *Journal of Machine Learning Research*, *15*, 2773–2832. https://jmlr.csail.mit.edu/papers/v15/anandkumar14b.html

Anandkumar A., Ge R., & Janzamin M. (2014c). Guaranteed non-orthogonal tensor decomposition via alternating rank-1 updates, arXiv, arXiv:1402.5180, preprint: not peer reviewed.

Auddy A., & Yuan M. (2023a). Large dimensional independent component analysis: Statistical optimality and computational tractability, arXiv, arXiv:2303.18156, preprint: not peer reviewed.

Auddy A., & Yuan M. (2023b). Perturbation bounds for (nearly) orthogonally decomposable tensors with statistical applications. *Information and Inference: A Journal of the IMA*, *12*(2), 1044–1072. https://doi.org/10.1093/imaiai/iaac033

Auffinger A., Arous G. B., & Černỳ J. (2013). Random matrices and complexity of spin glasses. *Communications on Pure and Applied Mathematics*, *66*(2), 165–201. https://doi.org/10.1002/cpa.v66.2

Bai J. (2003). Inferential theory for factor models of large dimensions. *Econometrica*, *71*(1), 135–171. https://doi.org/10.1111/ecta.2003.71.issue-1

Bai J., & Ng S. (2002). Determining the number of factors in approximate factor models. *Econometrica*, *70*(1), 191–221. https://doi.org/10.1111/ecta.2002.70.issue-1

Bai J., & Ng S. (2007). Determining the number of primitive shocks in factor models. *Journal of Business & Economic Statistics*, *25*(1), 52–60. https://doi.org/10.1198/073500106000000413

Bai J., & Wang P. (2014). Identification theory for high dimensional static and dynamic factor models. *Journal of Econometrics*, *178*(2), 794–804. https://doi.org/10.1016/j.jeconom.2013.11.001

Bai J., & Wang P. (2015). Identification and Bayesian estimation of dynamic factor models. *Journal of Business & Economic Statistics*, *33*(2), 221–240. https://doi.org/10.1080/07350015.2014.941467

Bekker P. (1986). A note on the identification of restricted factor loading matrices. *Psychometrika*, *51*(4), 607–611. https://doi.org/10.1007/BF02295600

Bi X., Qu A., & Shen X. (2018). Multilayer tensor factorization with applications to recommender systems. *The Annals of Statistics*, *46*(6B), 3308–3333. https://doi.org/10.1214/17-AOS1659

Box G., & Jenkins G. (1976). *Time series analysis, forecasting and control*. Holden Day.

Bradley R. C. (2005). Basic properties of strong mixing conditions. A survey and some open questions. *Probability Surveys*, *2*, 107–144. https://doi.org/10.1214/154957805100000104

Chamberlain G., & Rothschild M. (1983). Arbitrage, factor structure, and mean-variance analysis on large asset markets. *Econometrica*, *51*(5), 1281–1304. https://doi.org/10.2307/1912275

Chang J., He J., Yang L., & Yao Q. (2023). Modelling matrix time series via a tensor CP-decomposition. *Journal of the Royal Statistical Society Series B: Statistical Methodology*, *85*(1), 127–148. https://doi.org/10.1093/jrsssb/qkac011

Chen E. Y., & Fan J. (2023). Statistical inference for high-dimensional matrix-variate factor models. *Journal of the American Statistical Association*, *118*(542), 1038–1055. https://doi.org/10.1080/01621459.2021.1970569

Chen E. Y., Tsay R. S., & Chen R. (2020). Constrained factor models for high-dimensional matrix-variate time series. *Journal of the American Statistical Association*, *115*(530), 775–793. https://doi.org/10.1080/01621459.2019.1584899

Chen E. Y., Xia D., Cai C., & Fan J. (2024). Semi-parametric tensor factor analysis by iteratively projected singular value decomposition. *Journal of the Royal Statistical Society Series B: Statistical Methodology*, *86*(3), 793–823. https://doi.org/10.1093/jrsssb/qkae001

Chen R., Yang D., & Zhang C.-H. (2022). Factor models for high-dimensional tensor time series. *Journal of the American Statistical Association*, *117*(537), 94–116. https://doi.org/10.1080/01621459.2021.1912757

Comon P., Luciani X., & De Almeida A. L. (2009). Tensor decompositions, alternating least squares and other tales. *Journal of Chemometrics: A Journal of the Chemometrics Society*, *23*(7–8), 393–405. https://doi.org/10.1002/cem.v23:7/8

De Lathauwer L., De Moor B., & Vandewalle J. (2000). On the best rank-1 and rank-($r_1, r_2,\ldots, r_n$) approximation of higher-order tensors. *SIAM Journal on Matrix Analysis and Applications*, *21*(4), 1324–1342. https://doi.org/10.1137/S0895479898346995

Fan J., Liao Y., & Mincheva M. (2011). High dimensional covariance matrix estimation in approximate factor models. *The Annals of Statistics*, *39*(6), 3320. https://doi.org/10.1214/11-AOS944

Fan J., Liao Y., & Mincheva M. (2013). Large covariance estimation by thresholding principal orthogonal complements. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, *75*(4), 603–680. https://doi.org/10.1111/rssb.12016

Fan J., Liao Y., & Wang W. (2016). Projected principal component analysis in factor models. *The Annals of Statistics*, *44*(1), 219–254. https://doi.org/10.1214/15-AOS1364

Fan J., & Yao Q. (2003). *Nonlinear time series: Nonparametric and parametric methods*. Springer series in statistics. Springer-Verlag.

Forni M., Hallin M., Lippi M., & Reichlin L. (2000). The generalized dynamic-factor model: Identification and estimation. *The Review of Economics and Statistics*, *82*(4), 540–554. https://doi.org/10.1162/003465300559037

Forni M., Hallin M., Lippi M., & Reichlin L. (2004). The generalized dynamic factor model consistency and rates. *Journal of Econometrics*, *119*(2), 231–255. https://doi.org/10.1016/S0304-4076(03)00196-9

Forni M., Hallin M., Lippi M., & Reichlin L. (2005). The generalized dynamic factor model: One-sided estimation and forecasting. *Journal of the American Statistical Association*, *100*(471), 830–840. https://doi.org/10.1198/016214504000002050

Fosdick B. K., & Hoff P. D. (2014). Separable factor analysis with applications to mortality data. *Annals of Applied Statistics*, *8*(1), 120. https://doi.org/10.1214/13-AOAS694

Hallin M., & Liška R. (2007). Determining the number of factors in the general dynamic factor model. *Journal of the American Statistical Association*, *102*(478), 603–617. https://doi.org/10.1198/016214506000001275

Han R., Shi P., & Zhang A. R. (2023). Guaranteed functional tensor singular value decomposition. *Journal of the American Statistical Association*, 1–13. https://doi.org/10.1080/01621459.2022.2153689

Han Y., Chen R., Yang D., & Zhang C.-H. (2020). Tensor factor model estimation by iterative projection, arXiv, arXiv:2006.02611, preprint: not peer reviewed.
Han Y., Chen R., & Zhang C.-H. (2022). Rank determination in tensor factor model. *Electronic Journal of Statistics*, *16*(1), 1726–1803. https://doi.org/10.1214/22-EJS1991
Han Y., & Zhang C.-H. (2023). Tensor principal component analysis in high dimensional CP models. *IEEE Transactions on Information Theory*, *69*(2), 1147–1167. https://doi.org/10.1109/TIT.2022.3203972
Hao B., Zhang A., & Cheng G. (2020). Sparse and low-rank tensor estimation via cubic sketchings. *IEEE Transactions on Information Theory*, *66*(9), 5927–5964. https://doi.org/10.1109/TIT.18
Håstad J. (1990). Tensor rank is NP-complete. *Journal of Algorithms*, *11*(4), 644–654. https://doi.org/10.1016/0196-6774(90)90014-6
Hillar C. J., & Lim L. -H. (2013). Most tensor problems are NP-hard. *Journal of the ACM (JACM)*, *60*(6), 1–39. https://doi.org/10.1145/2512329
Hoff P. D. (2011). Separable covariance arrays via the tucker product, with applications to multivariate relational data. *Bayesian Analysis*, *6*(2), 179–196. https:/doi.org/10.1214/11-BA606
Hoff P. D. (2015). Multilinear tensor regression for longitudinal relational data. *Annals of Applied Statistics*, *9*(3), 1169. https://doi.org/10.1214/15-AOAS839
Huang B., Mu C., Goldfarb D., & Wright J. (2015). Provable models for robust low-rank tensor completion. *Pacific Journal of Optimization*, *11*(2), 339–364. http://www.yokohamapublishers.jp/online2/oppjo/vol11/p339.html.
Jain P., & Oh S. (2014). Learning mixtures of discrete product distributions using spectral decompositions. In *Conference on Learning Theory* (pp. 824–856). PMLR.
Kolda T. G., & Bader B. W. (2009). Tensor decompositions and applications. *SIAM Review*, *51*(3), 455–500. https://doi.org/10.1137/07070111X
Kuleshov V., Chaganty A., & Liang P. (2015). Tensor factorization via matrix factorization. In *Artificial intelligence and statistics* (pp. 507–516). PMLR.
Lam C., & Yao Q. (2012). Factor modeling for high-dimensional time series: Inference for the number of factors. *The Annals of Statistics*, *40*(2), 694–726. https://doi.org/10.1214/12-AOS970
Lam C., Yao Q., & Bathia N. (2011). Estimation of latent factors for high-dimensional time series. *Biometrika*, *98*(4), 901–918. https://doi.org/10.1093/biomet/asr048
Liu J., Musialski P., Wonka P., & Ye J. (2012). Tensor completion for estimating missing values in visual data. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *35*(1), 208–220. https://doi.org/10.1109/TPAMI.2012.39
Liu Y., Shang F., Fan W., Cheng J., & Cheng H. (2014). Generalized higher-order orthogonal iteration for tensor decomposition and completion. In Advances in neural information processing systems (pp. 1763–1771). Curran Associates.
Neudecker H. (1990). On the identification of restricted factor loading matrices: An alternative condition. *Journal of Mathematical Psychology*, *34*(2), 237–241. https://doi.org/10.1016/0022-2496(90)90004-S
Omberg L., Golub G. H., & Alter O. (2007). A tensor higher-order singular value decomposition for integrative analysis of dna microarray data from different studies. *Proceedings of the National Academy of Sciences*, *104*(47), 18371–18376. https://doi.org/10.1073/pnas.0709146104
Pan J., & Yao Q. (2008). Modelling multiple time series via common factors. *Biometrika*, *95*(2), 365–379. https://doi.org/10.1093/biomet/asn009
Pena D., & Box G. E. (1987). Identifying a simplifying structure in time series. *Journal of the American Statistical Association*, *82*(399), 836–843. https://doi.org/10.2307/2288794
Reinsel G. C. (2003). *Elements of multivariate time series analysis*. Springer.
Richard E., & Montanari A. (2014). A statistical model for tensor PCA. In Advances in neural information processing systems (Vol. 27). Curran Associate.
Rosenblatt M. (2012). *Markov processes, structure and asymptotic behavior: Structure and asymptotic behavior* (Vol. *184*). Springer Science & Business Media.
Sharan V., & Valiant G. (2017). Orthogonalized ALS: A theoretically principled tensor decomposition algorithm for practical use. In *International Conference on Machine Learning* (pp. 3095–3104). PMLR.
Shumway R., & Stoffer D. (2006). *Time series analysis and its applications: With R examples*. Springer.
Stock J. H., & Watson M. W. (2002). Forecasting using principal components from a large number of predictors. *Journal of the American Statistical Association*, *97*(460), 1167–1179. https://doi.org/10.1198/016214502388618960
Sun W. W., & Li L. (2017). STORE: Sparse tensor response regression and neuroimaging analysis. *Journal of Machine Learning Research*, *18*(1), 4908–4944. https://jmlr.org/papers/v18/17-203.html
Sun W. W., Lu J., Liu H., & Cheng G. (2017). Provable sparse tensor decomposition. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, *3*(79), 899–916. https://doi.org/10.1111/rssb.12190
Tong H. (1990). *Non-linear time series: A dynamical system approach*. Oxford University Press.
Tsay R. S. (2005). *Analysis of financial time series* (Vol. *543*). John Wiley & Sons.

Tsay R. S., & Chen R. (2018). *Nonlinear time series analysis* (Vol. *891*). John Wiley & Sons.
Wang D., Liu X., & Chen R. (2019). Factor models for matrix-valued high-dimensional time series. *Journal of Econometrics*, *208*(1), 231–248. https://doi.org/10.1016/j.jeconom.2018.09.013
Wang M., & Li L. (2020). Learning from binary multiway data: Probabilistic tensor decomposition and its statistical optimality. *Journal of Machine Learning Research*, *21*(154), 1–38. https://www.jmlr.org/papers/v21/18-766.html
Wang M., & Song Y. (2017). Tensor decompositions via two-mode higher-order SVD (HOSVD). In *Artificial intelligence and statistics* (pp. 614–622). PMLR.
Wang P.-A., & Lu C.-J. (2017). Tensor decomposition via simultaneous power iteration. In *International Conference on Machine Learning* (pp. 3665–3673). PMLR.
Zhang A., & Wang M. (2019). Spectral state compression of Markov processes. *IEEE Transactions on Information Theory*, *66*(5), 3202–3231. https://doi.org/10.1109/TIT.2019.2956737
Zhang A., & Xia D. (2018). Tensor SVD: Statistical and computational limits. *IEEE Transactions on Information Theory*, *64*(11), 7311–7338. https://doi.org/10.1109/TIT.2018.2841377
Zhou H., Li L., & Zhu H. (2013). Tensor regression with applications in neuroimaging data analysis. *Journal of the American Statistical Association*, *108*(502), 540–552. https://doi.org/10.1080/01621459.2013.776499
Zhu Z., Li X., Wang M., & Zhang A. (2022). Learning Markov models via low-rank optimization. *Operations Research*, *70*(4), 2384–2398. https://doi.org/10.1287/opre.2021.2115