Submitted to the Annals of Statistics

RANK AND FACTOR LOADINGS ESTIMATION IN TIME SERIES TENSOR FACTOR MODEL BY PRE-AVERAGING

BY WEILIN CHEN[1,a] AND CLIFFORD LAM[1,b]

[1]*Department of Statistics, London School of Economics and Political Science* , [a]*w.chen56@lse.ac.uk*; [b]*c.lam2@lse.ac.uk*

The idiosyncratic components of a tensor time series factor model can exhibit serial correlations, (e.g. finance or economic data), ruling out many state-of-the-art methods that assume white/independent idiosyncratic components. While the traditional higher order orthogonal iteration (HOOI) is proved to be convergent to a set of factor loading matrices, the closeness of them to the true underlying factor loading matrices are in general not established, or only under i.i.d. Gaussian noises. Under the presence of serial and cross-correlations in the idiosyncratic components and time series variables with only bounded fourth order moments, for tensor time series data with tensor order two or above, we propose a pre-averaging procedure that can be considered a random projection method. The estimated directions corresponding to the strongest factors are then used for projecting the data for a potentially improved re-estimation of the factor loading spaces themselves, with theoretical guarantees and rate of convergence spelt out when not all factors are pervasive. We also propose a new rank estimation method which utilizes correlation information from the projected data. Extensive simulations are performed and compared to other state-of-the-art or traditional alternatives. A set of tensor-valued NYC taxi data is also analyzed.

1. Introduction. Thanks to the advancement of the internet and general computing power, the collection and analysis of panel data are made ever easier over the past decade. Toolboxes in high dimensional vector time series analysis play increasingly important roles in extracting useful information from high dimensional time series data. Time series factor modelling is a major dimension reduction tool for such data, allowing insights into the common dynamics of different observed time series. For instance, when considering many macroeconomic time series for forecasting (Stock and Watson, 2002), the estimation and forecasting through the common factors can give more accurate results overall, and allowing for the interpretation of the factors (e.g., potential grouping of macroeconomic time series as factors) at the same time.

To improve the accuracy of forecasting, one can add the time series of macroeconomic indicators from other countries, and stack all observed time series into one high dimensional vector time series. The problem in doing this is that we are now ignoring the natural structure of the data, namely, all macroeconomic time series are now categorized by countries. Moreover, stacking all time series into a long vector can create curse of dimensionality (e.g., when the stacked length is too much larger than the sample size), leading to inaccurate estimation and predictions.

A more natural approach is to consider the country-categorized macroeconomic time series as matrix-valued (i.e., an *order-2 tensor*), with different countries by row and different macroeconomic time series by columns. Wang, Liu and Chen (2019) describes a factor model for such matrix-valued time series, and provides estimation methods together with theoretical

MSC2020 subject classifications: Primary 62H25, 62H12.

Keywords and phrases: Core rank tensor, tensor fibres pre-averaging, strongest factors projection, iterative projection algorithm, bootstrap tensor fibres.

results. Their work is extended to a general order-K tensor $\{\mathcal{X}_t\}$ in Chen, Yang and Zhang (2022), where the factor model for each $\mathcal{X}_t \in \mathbb{R}^{d_1 \times \cdots \times d_K}$, is

$$\mathcal{X}_t = \mathcal{C}_t + \mathcal{E}_t = \mathcal{F}_t \times_1 \mathbf{A}_1 \times_2 \cdots \times_K \mathbf{A}_K + \mathcal{E}_t, \tag{1.1}$$

with $\mathcal{C}_t$ the common component, $\mathcal{E}_t$ the noise tensor, $\mathcal{F}_t \in \mathbb{R}^{r_1 \times \cdots \times r_K}$ the core tensor, and $\mathbf{A}_k \in \mathbb{R}^{d_k \times r_k}$ the mode-k factor loading matrix. The product $\times_k$ is the tensor k-mode product (see Section 2 for a review of basic tensor operations). Chen, Yang and Zhang (2022) assumes that the elements in each $\mathcal{E}_t$ are sub-Gaussian, with each $\mathcal{E}_t$ independent of each other. Base on the above, Han et al. (2020) analyzes iterative projection procedures iTOPUP and iTIPUP for estimating $\mathbf{A}_k$, while Han, Zhang and Chen (2022a) proposes core rank (or multilinear tensor rank) estimators of $\mathcal{C}_t$ based on information criterion and eigen-ratio criterion that are intertwined with iTIPUP and iTOPUP. Core rank r_k is similar to the number of factors, and will be explained in Section 3.

In other recent developments, Zhang and Xia (2018) proposes a similar model for an order-3 tensor, with the tensor noise elements being i.i.d. normal having a common variance, and develops minimax theoretical guarantees for their estimators. With the same tensor noise assumption, Yokota, Lee and Cichocki (2017) proposes a core rank estimator for $\mathcal{C}_t$ for a general order-K tensor $\mathcal{X}_t$ based on a BIC-like criterion, while Liu, Yuan and Zhao (2022) proposes a tensor SVD method, and Han and Zhang (2023) proposes a tensor PCA for estimation under a CP decomposition of $\mathcal{C}_t$. Chen et al. (2020a) proposes a semiparametric model with $\mathcal{C}_t$ taking covariates under the assumption of i.i.d. sub-Gaussian elements in $\mathcal{E}_t$, which are themselves independent of each other.

All the tensor factor modelling works mentioned above assumed at least independent noise tensor series $\{\mathcal{E}_t\}$ with sub-Gaussian elements. The i.i.d. assumption for the elements in $\mathcal{E}_t$ in many of them is also considered a standard assumption for statistical analysis. However, if we have applications in economics and finance for instance, it is very easy that (weak) serial correlations exist in $\{\mathcal{E}_t\}$, representing any serial correlations in $\mathcal{X}_t$ not captured by the common components $\mathcal{C}_t$ (some time series in $\mathcal{X}_t$ have "unique" company or macroeconomic characteristics, for example). The *Approximate factor model* of Bai and Ng (2002) allows for such weak serial correlations (as well as weak cross-correlations) in the idiosyncratic noise series $\{\mathcal{E}_t\}$. When $\mathcal{E}_t$ has a higher order tensor structure, allowing for weak-serial and cross-correlations becomes even more essential as there could be even more potentially intricate serial and cross-correlations in $\{\mathcal{E}_t\}$. In this paper, we adopt such a more flexible approach. Our methods utilize covariance information, which are more natural to apply to financial return data for example as opposed to methods that utilize only autocovariance information (see Wang, Liu and Chen (2019), Chen, Yang and Zhang (2022) or Han, Zhang and Chen (2022b) for example). Due to market efficiency, population autocovariances of the data can be close to zero and methods that only utilize autocovariance information can have low signal-to-noise ratio.

For matrix factor models (i.e., an order-2 tensor) with weak-serial and cross-correlations in $\{\mathcal{E}_t\}$, Chen and Fan (2021) proposes an α-PCA method by assuming α-mixing of noise series, while He et al. (2022) proposes matrix Kendall's tau by assuming matrix elliptical distribution of the noise. With α-mixing assumption, Yu et al. (2022) develops a projection estimation (PE) method for matrix factor models by projecting the observation matrix onto the row or column factor space. The number of row and column factors are also estimated by the eigenvalue-ratio statistics based on the covariance information after projection. He et al. (2023a) provides the least squares interpretation of PE, and proposes a robust method by substituting the least squares loss function with the Huber Loss function (see also He et al. (2023b)). As an extension, He, Li and Trapani (2022) and Barigozzi et al. (2023) further generalize PE and the robust method to estimate tensor factor models for a general K. However, one limitation of all these recent developments is that they assume all factors are pervasive in every mode of the matrix or tensor, which can be restrictive in many real applications when weak factors are present.

Other related developments of tensor factor models include Liu and Chen (2022) on a threshold matrix-variate factor model, allowing general but uniform factor strengths on the factor loading matrices under uncorrelated noise tensor and an alpha-mixing condition for the factors. Chen et al. (2020b) proposes a class of models for modelling large-scale multivariate spatial-temporal processes, which involves known time-evolving covariates and a corresponding loading matrix, while all processes are dependent on a space domain. Chen et al. (2022) introduces a general R package `tensorTS` for wide variety of tensor data analyses based on recent papers, and demonstrate the usage on the NYC taxi data which we are analysing in Section 6.4.

In this paper, we make two important contributions to the literature of factor modelling for tensor time series data of order two or above. The first one is to allow for a spectrum of different factor strengths, which is a generalisation to Lam, Yao and Bathia (2011) when static vector time series factor model is concerned. To the best of our knowledge, our model is the first one in tensor factor modelling to consider weak factors when both serial and cross-correlations in $\{\mathcal{E}_t\}$ are present. For tensor factor models with independent $\{\mathcal{E}_t\}$, while Han et al. (2020) has two parameters δ_0 and δ_1 controlling the factor strengths, they are less easily interpretable compared to our $\alpha_{k,j}$, $j \in \{1, 2, \cdots, r_k\}$, which has the jth diagonal entry of $\mathbf{A}_k^{\mathrm{T}}\mathbf{A}_k \asymp d_k^{\alpha_{k,j}}$ (see Assumption (L1) in Section 3.2.3 for more details). Hence if the jth column of $\mathbf{A}_k$ is dense (a pervasive factor), then $\alpha_{k,j} = 1$. If there are only finitely many nonzeros in the jth column of $\mathbf{A}_k$, then it is a very weak factor, and $\alpha_{k,j} = 0$. Freyaldenhoven (2022) allows for these weaker factors in its vector time series factor model, and called them "local factors".

With relaxed assumptions for wider applications, and allowing for a spectrum of factors with different strengths, our second contribution is to provide a "pre-averaging" initial estimator and an iterative projection estimator for our model, with theoretical analyzes provided and rate of convergence spelt out. The pre-averaging procedure is presented in Section 3, which can be seen as a random projection method by randomly summing tensor fibres, and we provide a method to control for the quality of the random projection in Section 3.3. Section 3.6 also shows that our pre-averaging estimator is minimax optimal under certain scenarios on a certain localized set. Iterative projection estimators of the factor loading matrices (see Section 4) are provided with idea similar to the projection method in Yu et al. (2022), except that we only project on the direction aligning to the strongest estimated factor. This is because we assume there are weak factors which may not be estimated with enough accuracy. With weak factors in the model, numerical experiments show that our estimator outperforms other state-of-the-art methods since we only utilize the information which captures the most accurate estimations so far. To complete the paper, we also provide estimators of the core tensor rank through correlation analysis in Section 5, which is inspired by Fan, Guo and Zheng (2022), but we provide a bootstrap method for tuning parameter selection as well. All our methods are written into an R package `TensorPreAve` published on CRAN and GitHub. Please see Section 2 in our supplement for a very brief explanation on how to use it.

The rest of the paper is organized as follows. Section 2 reviews some basic notations we use throughout the paper. Section 3 presents the idea of pre-averaging, together with important assumptions on our model. Discussions and theory on choosing the "best" samples for aggregating results are presented, together with rate of convergence for our pre-averaging estimator for the strongest factors spelt out. Section 4 utilizes the pre-averaging estimator as the ideal initial estimator for re-estimating the projection direction by iterations, and presents the key theoretical results on the iterative projection estimators. Section 5 presents theoretical justifications for using correlation analysis in finding the rank of the core tensor, and provides a fibre bootstrapping technique in determining the tuning parameter of the procedure. Section 6 presents our simulation studies on a number of different settings and compare to other benchmarks or state-of-the-art estimators. A set of matrix-valued portfolio return data is analyzed in Section 1.3 in our supplement, and a tensor-valued NYC taxi data set is analyzed in Section 6.4 of this paper. All proofs are in Section 3 of our supplement.

2. Notations and Basic Tensor Manipulations.

In this paper, we use $a \asymp b$ to denote $a = O(b)$ and $b = O(a)$ (also $a \asymp_P b$ for $a = O_P(b)$ and $b = O_P(a)$), while $a \succeq b$ is equivalent to $b = O(a)$, and $a \succ b$ is equivalent to $b = o(a)$. We also use $\|\cdot\|$ to denote the L_2 norm (of a vector or a matrix), and $\|\cdot\|_F$ to denote the Frobenius norm, while $\|\cdot\|_{\max}$ represents the maximum element (of a vector or a matrix). We also use $\|\mathbf{A}\|_\infty = \max_i \sum_j |a_{ij}|$ and $\|\mathbf{A}\|_1 = \max_j \sum_i |a_{ij}|$ to denote the L_∞ and L_1 norm of a matrix $\mathbf{A}$ respectively. The notation vec$(\cdot)$ represents the vectorisation of a matrix, stacking columns of the matrix from left to right. We use $\mathbf{1}_m$ to represent a vector of ones with length m, $\mathbf{1}_{m,S}$ to represent a vector of ones and zeros with length m, with ones on positions belonging to the set S and zeros otherwise. The identity matrix with size m is denoted by $\mathbf{I}_m$. The notation diag$(\mathbf{A})$ of a square matrix $\mathbf{A}$ is the diagonal matrix with only the diagonal elements of $\mathbf{A}$ remain, and everything else set to 0. This notation is also used to represent a block diagonal matrix. For instance, diag$(\mathbf{A}_1, \ldots, \mathbf{A}_n)$ is the block diagonal matrix with diagonal block matrices $\mathbf{A}_1, \ldots, \mathbf{A}_n$. We use $\lambda_j(\mathbf{A})$ to denote the j-th largest eigenvalue of a square matrix $\mathbf{A}$, and tr$(\mathbf{A})$ the trace of $\mathbf{A}$. For a positive integer m, we define $[m] := \{1, \ldots, m\}$. The cardinality of a set S is denoted by $|S|$.

We briefly introduce the notations and review on tensor manipulations in this section just enough for the presentation of this paper. For more information, please refer to Kolda and Bader (2009).

Let $\mathcal{X} \in \mathbb{R}^{d_1 \times \cdots \times d_K}$ be an order-K tensor. Here K represents the number of dimensions in $\mathcal{X}$, also called the number of *modes*. For instance, a vector time series has $K = 1$ while a matrix time series has $K = 2$. If we write $\mathcal{X} = (x_{i_1 \cdots i_K})$, then we define a *mode-k fibre* of $\mathcal{X}$ to be a column vector (of length d_k) $(x_{i_1 \cdots i_{k-1}, j, i_{k+1} \cdots i_K})_{j \in [d_k]}$, $i_\ell \in [d_\ell]$ with $\ell \in [K]$. Hence there are in total $d_{\text{-}k} := \prod_{\ell=1\,;\ell \neq k}^K d_\ell$ number of mode-k fibres for the tensor $\mathcal{X}$. The *mode-k matricization/unfolding matrix* mat$_k(\mathcal{X}) \in \mathbb{R}^{d_k \times d_{\text{-}k}}$ (also denoted as $\mathbf{X}_{(k)}$ sometimes) is then defined to be the matrix containing (in order) all the mode-k fibres of $\mathcal{X}$. See figure 1 for a demonstration (figure from Tao, Su and Wang (2019)).

If there is a matrix $\mathbf{A} \in \mathbb{R}^{I_k \times r_k}$, and $\mathcal{F} \in \mathbb{R}^{r_1 \times \cdots r_K}$ is an order-K tensor, then the *k-mode product* of $\mathcal{F}$ and $\mathbf{A}$, denoted by $\mathcal{F} \times_k \mathbf{A} \in \mathbb{R}^{r_1 \times \cdots \times r_{k-1} \times I_k \times r_{k+1} \times \cdots \times r_K}$, is defined such that mat$_k(\mathcal{F} \times_k \mathbf{A}) = \mathbf{A}mat_k(\mathcal{F})$. The order of distinct mode products does not matter, in the sense that for $i \neq j$, $\mathcal{F} \times_i \mathbf{A}_i \times_j \mathbf{A}_j = \mathcal{F} \times_j \mathbf{A}_j \times_i \mathbf{A}_i$. Finally, if $\mathcal{C} = \mathcal{F} \times_1 \mathbf{A}_1 \times_2 \cdots \times_K \mathbf{A}_K$, then we have the formula

$$\text{mat}_k(\mathcal{C}) = \mathbf{A}_k \text{mat}_k(\mathcal{F}) \mathbf{A}_{\text{-}k}^{\mathrm{T}}, \tag{2.1}$$

where $\otimes$ is the Kronecker product, and $\mathbf{A}_{\text{-}k} := \mathbf{A}_K \otimes \cdots \otimes \mathbf{A}_{k+1} \otimes \mathbf{A}_{k-1} \otimes \cdots \otimes \mathbf{A}_1$.

3. Initial Estimation of Strongest Factors by Pre-averaging.

We define the tensor factor model for each $\mathcal{X}_t \in \mathbb{R}^{d_1 \times \cdots \times d_K}$, $t \in [T]$, as

$$\mathcal{X}_t = \mu + \mathcal{C}_t + \mathcal{E}_t = \mu + \mathcal{F}_t \times_1 \mathbf{A}_1 \times_2 \cdots \times_K \mathbf{A}_K + \mathcal{E}_t, \tag{3.1}$$

where we include a non-zero mean tensor $\mu \in \mathbb{R}^{d_1 \times \cdots \times d_K}$ as compared to (1.1) introduced by Chen, Yang and Zhang (2022), which makes our model more flexible. Using (2.1), the mode-k unfolding of (3.1) can be written as

$$\text{mat}_k(\mathcal{X}_t) = \text{mat}_k(\mu) + \mathbf{A}_k \text{mat}_k(\mathcal{F}_t) \mathbf{A}_{\text{-}k}^{\mathrm{T}} + \text{mat}_k(\mathcal{E}_t).$$

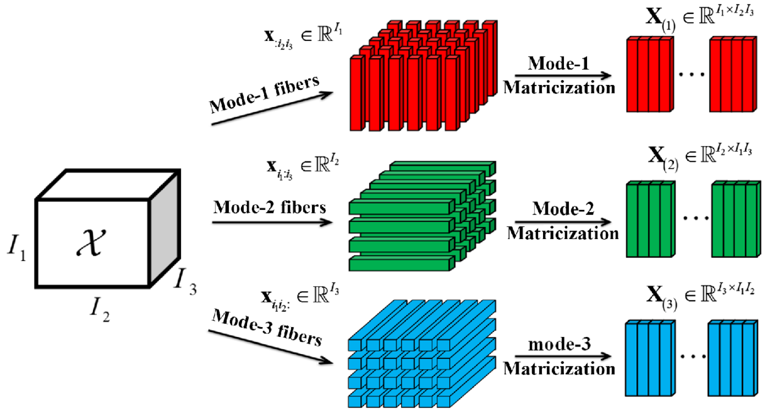

I_1
$\mathcal{X}$
I_2
I_3
$\mathbf{x}_{:i_2i_3} \in \mathbb{R}^{I_1}$
Mode-1 fibers
Mode-1 Matricization
$\mathbf{X}_{(1)} \in \mathbb{R}^{I_1 \times I_2 I_3}$
$\mathbf{x}_{i_1:i_3} \in \mathbb{R}^{I_2}$
Mode-2 fibers
Mode-2 Matricization
$\mathbf{X}_{(2)} \in \mathbb{R}^{I_2 \times I_1 I_3}$
$\mathbf{x}_{i_1i_2:} \in \mathbb{R}^{I_3}$
Mode-3 fibers
mode-3 Matricization
$\mathbf{X}_{(3)} \in \mathbb{R}^{I_3 \times I_1 I_2}$

Fig 1: Illustration of the mode-k fibers and its corresponding unfolding matrix.

If we define $S_j \subseteq [d_j]$ for $j \in [K]$, then we can always define the Cartesian product

$$S_{\text{-}k} := S_K \times \cdots \times S_{k+1} \times S_{k-1} \times \cdots \times S_1, \text{ such that}$$

$$\mathbf{1}_{d_{\text{-}k}, S_{\text{-}k}} = \mathbf{1}_{d_K, S_K} \otimes \cdots \otimes \mathbf{1}_{d_{k+1}, S_{k+1}} \otimes \mathbf{1}_{d_{k-1}, S_{k-1}} \otimes \cdots \otimes \mathbf{1}_{d_1, S_1}.$$

Projecting on $\mathbf{1}_{d_{\text{-}k}, S_{\text{-}k}}$, equivalent to summing the fibres in $\text{mat}_k(\mathcal{X}_t)$ over the set $S_{\text{-}k}$, is then

$$\text{mat}_k(\mathcal{X}_t)\mathbf{1}_{d_{\text{-}k}, S_{\text{-}k}} = \text{mat}_k(\mu)\mathbf{1}_{d_{\text{-}k}, S_{\text{-}k}} + \mathbf{A}_k \text{mat}_k(\mathcal{F}_t)\mathbf{A}_{\text{-}k}^{\text{T}}\mathbf{1}_{d_{\text{-}k}, S_{\text{-}k}} + \text{mat}_k(\mathcal{E}_t)\mathbf{1}_{d_{\text{-}k}, S_{\text{-}k}}, \text{ where}$$

(3.2)

$$\mathbf{A}_{\text{-}k}^{\text{T}}\mathbf{1}_{d_{\text{-}k}, S_{\text{-}k}} = \mathbf{A}_K^{\text{T}}\mathbf{1}_{d_K, S_K} \otimes \cdots \otimes \mathbf{A}_{k+1}^{\text{T}}\mathbf{1}_{d_{k+1}, S_{k+1}} \otimes \mathbf{A}_{k-1}^{\text{T}}\mathbf{1}_{d_{k-1}, S_{k-1}} \otimes \cdots \otimes \mathbf{A}_1^{\text{T}}\mathbf{1}_{d_1, S_1},$$

with $\mathbf{A}_{\text{-}k} := \mathbf{A}_K \otimes \cdots \mathbf{A}_{k+1} \otimes \mathbf{A}_{k-1} \otimes \cdots \otimes \mathbf{A}_1$. Hence projection of the data using $\mathbf{1}_{d_{\text{-}k}, S_{\text{-}k}}$ can be seen as pre-averaging the rows of each $\mathbf{A}_j$ using S_j for $j \in [K] \setminus \{k\}$. While we re-estimate by projection in Section 4, and papers like Yu et al. (2022) does projection estimation as well, the aim of this section is to provide an initial estimator of projection direction with quality that can be *controlled* by careful selection of randomly generated S_j. The method to select S_j among multiple random samples is introduced in Section 3.3, which leads to the pre-averaging estimator in Section 3.5.

3.1. *Potential advantages of pre-averaging.* Consider just calculating the second order moments

$$\sum_{t=1}^{T} \text{mat}_k(\mathcal{X}_t - \bar{\mathcal{X}})\text{mat}_k^{\text{T}}(\mathcal{X}_t - \bar{\mathcal{X}}) =: S_0 + N_1 + N_1^{\text{T}} + N_2, \text{ where}$$

(3.3)

$$S_0 := \mathbf{A}_k \sum_{t=1}^{T} \Big(\text{mat}_k(\mathcal{F}_t - \bar{\mathcal{F}})\mathbf{A}_{\text{-}k}^{\text{T}}\mathbf{A}_{\text{-}k}\text{mat}_k^{\text{T}}(\mathcal{F}_t - \bar{\mathcal{F}})\Big)\mathbf{A}_k^{\text{T}},$$

$$N_1 := \mathbf{A}_k \sum_{t=1}^{T} \Big(\text{mat}_k(\mathcal{F}_t - \bar{\mathcal{F}})\mathbf{A}_{\text{-}k}^{\text{T}}\text{mat}_k^{\text{T}}(\mathcal{E}_t - \bar{\mathcal{E}})\Big), \quad N_2 := \sum_{t=1}^{T} \text{mat}_k(\mathcal{E}_t - \bar{\mathcal{E}})\text{mat}_k^{\text{T}}(\mathcal{E}_t - \bar{\mathcal{E}}),$$

and extracting an estimator of $\mathbf{A}_k$ through PCA (e.g., see Bai and Ng (2002)). Our proposed pre-averaging estimator, like a projected estimator, can accumulate significantly more signals before doing the PCA step for extracting an estimator of $\mathbf{A}_k$. This is because the signal term $\mathbf{A}_k \sum_{t=1}^{T} \text{mat}_k(\mathcal{F}_t - \bar{\mathcal{F}})\mathbf{A}_{\text{-}k}^{\text{T}}\mathbf{1}_{d_{\text{-}k}, S_{\text{-}k}}\mathbf{1}_{d_{\text{-}k}, S_{\text{-}k}}^{\text{T}}\mathbf{A}_{\text{-}k}\text{mat}_k^{\text{T}}(\mathcal{F}_t - \bar{\mathcal{F}})\mathbf{A}_k^{\text{T}}$ (from using $\mathbf{1}_{d_{\text{-}k}, S_{\text{-}k}}$ as the projection direction of the data) can be significantly larger than S_0 in (3.3), since the diagonal elements of $\mathbf{A}_{\text{-}k}^{\mathrm{T}}\mathbf{1}_{d_{\text{-}k},S_{\text{-}k}}\mathbf{1}_{d_{\text{-}k},S_{\text{-}k}}^{\mathrm{T}}\mathbf{A}_{\text{-}k}$ can be much larger than those in $\mathbf{A}_{\text{-}k}^{\mathrm{T}}\mathbf{A}_{\text{-}k}$. For instance, when $\mathbf{A}_{\text{-}k}$ has a column with all positive or negative elements (e.g., factor loading entries for the market factors in finance), we have diagonal elements of $\mathbf{A}_{\text{-}k}^{\mathrm{T}}\mathbf{1}_{d_{\text{-}k},S_{\text{-}k}}\mathbf{1}_{d_{\text{-}k},S_{\text{-}k}}^{\mathrm{T}}\mathbf{A}_{\text{-}k}$ of order $d_{\text{-}k}^2$, while those in $\mathbf{A}_{\text{-}k}^{\mathrm{T}}\mathbf{A}_{\text{-}k}$ are only of order $d_{\text{-}k}$.

Before officially introducing the pre-averaging estimator, we first present some technical assumptions needed for the tensor factor model (3.1).

3.2. *Assumptions.*

3.2.1. *Assumptions on the errors.* We present assumptions (E1) - (E2) below with explanations.

(E1) (Decomposition of error) *We assume that K is a constant, and*

$$\mathcal{E}_t = \mathcal{F}_{e,t} \times_1 \mathbf{A}_{e,1} \times \cdots \times \mathbf{A}_{e,K} + \boldsymbol{\epsilon}_t, \tag{3.4}$$

where $\mathcal{F}_{e,t}$ is an order-K tensor with dimension $r_{e,1} \times \cdots \times r_{e,K}$, containing independent elements with mean 0 and variance 1. The order-K tensor $\boldsymbol{\epsilon}_k \in \mathbb{R}^{d_1 \times \cdots \times d_K}$ contains independent mean zero elements each with finite variance, with the two time series $\{\boldsymbol{\epsilon}_t\}$ and $\{\mathcal{F}_{e,t}\}$ being independent.

Moreover, for each $k \in [K]$, $\mathbf{A}_{e,k} \in \mathbb{R}^{d_k \times r_{e,k}}$ is such that $\left\|\mathbf{A}_{e,k}\right\|_1 = O(1)$. That is, $\mathbf{A}_{e,k}$ is (approximately) sparse.

Hence with (E1), we have $\text{mat}_k(\mathcal{E}_t) = \mathbf{A}_{e,k}\text{mat}_k(\mathcal{F}_{e,t})\mathbf{A}_{e,\text{-}k}^{\mathrm{T}} + \text{mat}_k(\boldsymbol{\epsilon}_t)$, where $\mathbf{A}_{e,\text{-}k} := \mathbf{A}_{e,K} \otimes \cdots \otimes \mathbf{A}_{e,k+1} \otimes \mathbf{A}_{e,k-1} \otimes \cdots \otimes \mathbf{A}_{e,1}$. Each mode-$k$ noise fibre $\mathbf{e}_{t,\text{-}k,\ell}$ for $\ell \in [d_{\text{-}k}]$ can then be decomposed as

$$\mathbf{e}_{t,\text{-}k,\ell} := \mathbf{A}_{e,k}\text{mat}_k(\mathcal{F}_{e,t})\mathbf{a}_{e,\text{-}k,\ell} + (\boldsymbol{\Sigma}_{\epsilon,\ell}^{(k)})^{1/2}\boldsymbol{\epsilon}_{t,\ell}^{(k)}, \tag{3.5}$$

where $\mathbf{a}_{e,\text{-}k,\ell}^{\mathrm{T}}$ is the ℓ-th row of $\mathbf{A}_{e,\text{-}k}$, $\boldsymbol{\Sigma}_{\epsilon,\ell}^{(k)}$ is diagonal and $\boldsymbol{\epsilon}_{t,\ell}^{(k)}$ contains independent elements each with mean 0 and variance 1. The above decomposition means that each noise fibre is now a sum of two parts. The first part is similar to a common component with a factor loading matrix $\mathbf{A}_{e,k}$, while the second part contains independent noise (but can still exhibit serial correlations; see Assumption (E2)). However, $\mathbf{A}_{e,k}$ is (approximately) sparse here and contains at most a very weak factor with factor strength 0 (see Assumption (L1) in Section 3.2.3). This part facilitates cross-noise fibres correlations, with

$$\text{cov}(\mathbf{e}_{t,\text{-}k,\ell}, \mathbf{e}_{t,\text{-}k,m}) = \mathbf{a}_{e,\text{-}k,\ell}^{\mathrm{T}}\mathbf{a}_{e,\text{-}k,m}\mathbf{A}_{e,k}\mathbf{A}_{e,k}^{\mathrm{T}}.$$

This error structure satisfies the assumptions in He, Li and Trapani (2022) and Chen and Fan (2021) when $r_{e,k} = O(d_k)$, but we only assume up to the 4th order moments of the noise variables exist and that these moments are uniformly bounded in Assumption (R2), which is more relaxed than requiring the existent of 8th order moments in He, Li and Trapani (2022). In fact, if $\mathbf{A}_{e,k}$ is not (approximately) sparse, it should be counted as a factor loading matrix rather than a noise component in our model.

(E2) (Time series) *There is $\mathcal{Z}_{e,t}$ the same dimension as $\mathcal{F}_{e,t}$, and $\mathcal{Z}_{\epsilon,t}$ the same dimension as $\boldsymbol{\epsilon}_t$, such that $\mathcal{F}_{e,t} = \sum_{q\geq 0} a_{e,q}\mathcal{Z}_{e,t-q}$ and $\boldsymbol{\epsilon}_t = \sum_{q\geq 0} a_{\epsilon,q}\mathcal{Z}_{\epsilon,t-q}$, with $\{\mathcal{Z}_{e,t}\}$ and $\{\mathcal{Z}_{\epsilon,t}\}$ independent of each other, and each time series have i.i.d. elements with mean 0 and variance 1. The coefficients $a_{e,q}$ and $a_{\epsilon,q}$ are so that $\sum_{q\geq 0} a_{e,q}^2 = \sum_{q\geq 0} a_{\epsilon,q}^2 = 1$ and $\sum_{q\geq 0} |a_{e,q}| \leq C$, $\sum_{q\geq 0} |a_{\epsilon,q}| \leq C$ for some constant C.*

With this assumption, the error variables in $\mathcal{F}_{e,t}$ and $\boldsymbol{\epsilon}_t$ are serially correlated in general. Together with (E1), (weak) serial and cross-sectional dependence within and among fibres are allowed for the errors.

3.2.2. *Assumptions on the factors.* Similar to (E2), the factors in $\mathcal{F}_t$ are assumed to follow general linear processes.

(F1) *There is $\mathcal{Z}_{f,t}$ the same dimension as $\mathcal{F}_t$, such that $\mathcal{F}_t = \sum_{q\geq 0} a_{f,q}\mathcal{Z}_{f,t-q}$. The time series $\{\mathcal{Z}_{f,q}\}$ has i.i.d. elements with mean 0 and variance 1. The coefficients $a_{f,q}$ are so that $\sum_{q\geq 0} a_{f,q}^2 = 1$ and $\sum_{q\geq 0}|a_{f,q}| \leq C$ for some constant C.*

Note the series of coefficients $\{a_{e,q}\}$, $\{a_{\epsilon,q}\}$ and $\{a_{f,q}\}$ are not necessarily equal.

3.2.3. *Assumptions on the model parameters.* We present the assumptions needed for the factor loading matrices $\mathbf{A}_k$, $k \in [K]$, and other model parameters.

(L1) (Factor Strength) *We assume that, for $k \in [K]$, $\mathbf{A}_k$ is of full rank, $r_k = o(T^{1/3})$, and as $d_k \to \infty$,*

$$\mathbf{D}_k^{-1/2}\mathbf{A}_k^{\mathrm{T}}\mathbf{A}_k\mathbf{D}_k^{-1/2} \to \Sigma_{\mathbf{A},k}, \tag{3.6}$$

where $\mathbf{D}_k = \mathrm{diag}(\mathbf{A}_k^{\mathrm{T}}\mathbf{A}_k)$ and $\Sigma_{\mathbf{A},k}$ is positive definite with all eigenvalues bounded away from 0 and infinity. We assume $(\mathbf{D}_k)_{jj} \asymp d_k^{\alpha_{k,j}}$ for $j \in [r_k]$, and $0 < \alpha_{k,r_k} \leq \cdots \leq \alpha_{k,2} \leq \alpha_{k,1} \leq 1$.

Assumption (L1) states that the factors can have different strengths. When $K = 1$ and $\alpha_{1,j} = \alpha$ for $j \in [r_1]$, (3.6) reduces to the assumption of (approximate) vector factor model with the same strength, which is discussed in Bai and Ng (2021). Hence, our assumption is a generalisation of Bai and Ng (2021) to a tensor setting with mixed strengths of factors, which is more flexible to apply on many real datasets. In addition, we do not assume the orthogonality of $\mathbf{A}_k$ as Freyaldenhoven (2022) did, since this would be incompatible with the expression of factor strength and signal accumulation in terms of the norm and row sum of $\mathbf{A}_k$. The concept of a pervasive factor, for instance, depends on a column of $\mathbf{A}_k$ being dense. However, such an interpretation can be lost completely under the assumption of orthogonal columns in $\mathbf{A}_k$.

(L2) (Signal accumulation from summing) *For $k \in [K]$, let $M_{k,0} > 0$ be the number of different sums of rows of $\mathbf{A}_k$ considered, and for $m \in [M_{k,0}]$, denote $S_{k,m} \subseteq [d_k]$ to be the m-th index set for summing the rows of $\mathbf{A}_k$. With the choice of $|S_{k,m}| = \lfloor d_k/2 \rfloor$, define*

$$s_{k,m} := \left\|\mathbf{A}_k^{\mathrm{T}}\mathbf{1}_{d_k,S_{k,m}}\right\|^2, \quad s_{k,\max} := \max_{m\in[M_{k,0}]} s_{k,m}, \quad s_{\text{-}k,\max} := \prod_{j=1;j\neq k}^{K} s_{j,\max}. \tag{3.7}$$

We assume for some $z_k \leq r_k$,

$$\frac{d_{\text{-}k}}{s_{\text{-}k,\max}}\left(1+\frac{d_k}{T}\right) = o\left(d_k^{\alpha_{k,z_k}}\right). \tag{3.8}$$

In Assumption (L2), $s_{k,m}$ can be seen as a measure of accumulation of signals for a specific sample $m \in [M_{k,0}]$, and $s_{k,max}$ is the “largest” accumulation of signal we can attain over the $M_{k,0}$ samples. In Section 3.3, the method to provide a carefully selection of randomly generated $S_{j,m}$ is introduced, and Section 3.4 gives a more thorough discussion on the number of samples needed to secure enough signal accumulation with a large probability.

Note that we choose $S_{k,m}$ with size $|S_{k,m}| \asymp d_k$ (e.g., $|S_{k,m}| = \lfloor d_k/2 \rfloor$ in Assumption (L2)) for each $m \in [M_{k,0}]$. This choice allows for the sum of rows of $\mathbf{A}_k$ to be potentially large with a large probability (see also Section 3.4).

We also remark that unlike for instance in Chen and Fan (2021) that the dimensions d_k are assumed to be diverging, here d_k can be finite as long as $d_{\text{-}k}/s_{\text{-}k,\max} = o(1)$. This can be achieved when, for example, there is an $\mathbf{A}_j$ for some $j \neq k$ such that the majority of the elements in a column are of the same sign, so that $s_{j,m} \succ d_j$, resulting in $d_{\text{-}k}/s_{\text{-}k,\max} = o(1)$.

(R1) *The time series $\{\mathcal{Z}_{f,t}\}$ from Assumption (F1), $\{\mathcal{Z}_{e,t}\}$ and $\{\mathcal{Z}_{\epsilon,t}\}$ from Assumption (E2) are mutually independent of each other. All three time series have elements with uniformly bounded fourth moments.*

(R2) *We assume $\lambda_{d_k}(\boldsymbol{\Sigma}_{\epsilon,\ell}^{(k)})$ is uniformly bounded below from 0 for $\ell \in [d_{\text{-}k}]$, where $\boldsymbol{\Sigma}_{\epsilon,\ell}^{(k)}$ is defined in (3.5). Let $\mathbf{A}_{\epsilon,T}$ be the $T \times T$ matrix with its (t,s) element to be $(\mathbf{A}_{\epsilon,T})_{t,s} = \sum_{q\geq 0} a_{\epsilon,q}a_{\epsilon,q+|t-s|}$. Denote $0 < y := \lim_{d_k,T\to\infty} \frac{\min(d_k,T)}{\max(d_k,T)} \leq 1$ and $y^* = \min(y,1)$, then we assume there exists $c_1 \in (1-y^*, 1]$ such that $\lambda_{\lfloor c_1 T\rfloor}(\mathbf{A}_{\epsilon,T}) > c_2 > 0$ for large T, where c_2 is a positive constant.*

Assumption (R1) relaxes the need for Gaussian or sub-Gaussian random variables (see Zhang and Xia (2018) and Chen, Yang and Zhang (2022) for example), with only bounded fourth order moments required. This allows for substantially more types of data to be analyzed. For instance, financial returns data over more volatile periods where we do not usually want to assume moments beyond order four exist. Finally, together with Assumption (R1), Assumption (R2) enables us to utilize random matrix theory to bound the eigenvalues of various sample covariance matrices from below (see (S.3), (S.4) and (S.11) in Lemma 1 and Lemma 2 in our supplement). As long as the serial correlations of the $\epsilon_{t,\ell,j}^{(k)}$'s are not too strong, Assumption (R2) will be satisfied.

For convenience of further theoretical analysis, we define $\mathbf{Q}_k = \mathbf{A}_k\mathbf{D}_k^{-1/2}$. Since $\mathbf{Q}_k^{\mathrm{T}}\mathbf{Q}_k \to \Sigma_{A,k}$, $\mathbf{Q}_k$ is a re-normalized version of $\mathbf{A}_k$. In addition, we apply the singular value decomposition of $\mathbf{A}_k$ as

$$\mathbf{A}_k = \mathbf{U}_k\mathbf{G}_k^{1/2}\mathbf{V}_k^{\mathrm{T}}, \tag{3.9}$$

where $\mathbf{U}_k \in \mathbb{R}^{d_k\times r_k}$ has orthogonal columns such that $\mathbf{U}_k^{\mathrm{T}}\mathbf{U}_k = \mathbf{I}_{r_k}$, $\mathbf{G}_k \in \mathbb{R}^{r_k\times r_k}$ is diagonal and consists of the eigenvalues of $\mathbf{A}_k^{\mathrm{T}}\mathbf{A}_k$ in decreasing order, and $\mathbf{V}_k \in \mathbb{R}^{r_k\times r_k}$ is an orthogonal matrix. The subspaces spanned by the columns of $\mathbf{U}_k$, $\mathbf{Q}_k$ and $\mathbf{A}_k$ are the same, and hence it is equivalent to estimate $\mathbf{U}_k$ (or $\mathbf{Q}_k$) and $\mathbf{A}_k$, and the columns of $\mathbf{U}_k$ form an orthonormal basis for the column space spanned by $\mathbf{Q}_k$ (or $\mathbf{A}_k$). We will estimate $\mathbf{U}_k$ (or $\mathbf{Q}_k$) instead of $\mathbf{A}_k$ in the sections that follow. We need another regularity condition on the singular values on $\mathbf{G}_k$. This can be relaxed at the expense of lengthier explanations involving factor loading spaces in all subsequent theorems.

(L1') The singular values on $\mathbf{G}_k$ are distinct.

3.3. *Choosing samples of tensor fibres.*

We first present an algorithm for choosing the "best" sample of tensor fibres to sum.

Algorithm for choosing the "best" sample of tensor fibres

1. Initialize $M_{k,0}$ for each $k \in [K]$.
2. Generate a sequence of independent sets $\{S_{k,m}\}_{k\in[K],m\in[M_{k,0}]}$. Each $S_{k,m}$ chooses uniformly over $[d_k]$, with $|S_{k,m}| = \lfloor d_k/2\rfloor$.
3. Fix $k \in [K]$. Define $M_0 := \prod_{j\in[K]\setminus\{k\}} M_{j,0}$. For each $m \in [M_0]$, define $S_{\text{-}k,m} := \times_{j\in[K]\setminus\{k\}} S_{j,m_j}$ and $\mathbf{1}_{d_{\text{-}k},S_{\text{-}k,m}} := \otimes_{j\in[K]\setminus\{k\}} \mathbf{1}_{d_j,S_{j,m_j}}$ for some $m_j \in [M_{j,0}]$.
4. For the same fixed k from step 3, define for each $m \in [M_0]$,

$$\widetilde{\mathbf{X}}_{k,m} := (\mathrm{mat}_k(\mathcal{X}_1)\mathbf{1}_{d_{\text{-}k},S_{\text{-}k,m}}, \ldots, \mathrm{mat}_k(\mathcal{X}_T)\mathbf{1}_{d_{\text{-}k},S_{\text{-}k,m}})^{\mathrm{T}}, \tag{3.10}$$

and for an integer l satisfying $r_k + 1 \leq l \leq \lfloor c\min(T,d_k)\rfloor - r_k$ for some $c > 0$, construct

$$\mathrm{ER}_{l,m} := \frac{\lambda_1\big(\widetilde{\mathbf{X}}_{k,m}^{\mathrm{T}}\left(\mathbf{I}_T - \frac{1}{T}\mathbf{1}_T\mathbf{1}_T^{\mathrm{T}}\right)\widetilde{\mathbf{X}}_{k,m}\big)}{\lambda_l\big(\widetilde{\mathbf{X}}_{k,m}^{\mathrm{T}}\left(\mathbf{I}_T - \frac{1}{T}\mathbf{1}_T\mathbf{1}_T^{\mathrm{T}}\right)\widetilde{\mathbf{X}}_{k,m}\big)}.$$

5. The "best" sample $m \in [M_0]$ for estimating $\mathbf{A}_k$ is the one that maximizes $\mathrm{ER}_{l,m}$. We denote by $S_{\text{-}k,\max} := \times_{j\in[K]\setminus\{k\}} S_{j,\max}$ the corresponding product set, and
$$s_{\text{-}k,\max} := \prod_{j\in[K]\setminus\{k\}} s_{j,\max} := \prod_{j\in[K]\setminus\{k\}} \big\|\mathbf{A}_j^{\mathrm{T}}\mathbf{1}_{d_j,S_{j,\max}}\big\|^2.$$
6. Repeat steps 3,4,5 until each $k\in[K]$ is covered.

The justification of step 4 is as follows. With Assumption (L1) and (R2) satisfied, we have by Lemma 2 in our supplement that the eigenvalue-ratio $\mathrm{ER}_{l,m}$ has
$$\mathrm{ER}_{l,m} \asymp_P d_k^{\alpha_{k,1}}\left[\frac{d_{\text{-}k}}{s_{\text{-}k,m}}\left(1+\frac{d_k}{T}\right)\right]^{-1}.$$
Hence the sample that maximised $\mathrm{ER}_{l,m}$ in fact asymptotically maximizes the product of signals, from $s_{\text{-}k,m}$ to $s_{\text{-}k,\max}$.

One way to choose l is to use expert opinion. A typical value of l we use depends on the user's idea of the maximum value of r_k. Suppose for an economic data set, we expect $r_k \le 8$. Then we can use $l = 9$ for constructing ER_l. For a more data-driven way, note from Lemma 2 in our supplement that for a particular sample with product set $S_{\text{-}k,m} \subseteq [d_{\text{-}k}]$,
$$\lambda_i\Big(\widetilde{\mathbf{X}}_{k,m}^{\mathrm{T}}\Big(\mathbf{I}_T - \frac{1}{T}\mathbf{1}_T\mathbf{1}_T^{\mathrm{T}}\Big)\widetilde{\mathbf{X}}_{k,m}/T\Big) \asymp_P \begin{cases} d_k^{\alpha_{k,i}}, & i\in[r_k];\\ \frac{d_{\text{-}k}}{s_{\text{-}k,m}}\Big(1+\frac{d_k}{T}\Big), & r_k+1\le i\le \lfloor c\min(T,d_k)\rfloor - r_k,\end{cases}$$
where $s_{\text{-}k,m}$ is defined in (3.7), and $\widetilde{\mathbf{X}}_{k,m}$ in (3.10). Hence for $d_k \asymp T$, if we have a sample $S_{\text{-}k,m}$ such that $d_{\text{-}k}/s_{\text{-}k,m} = O(1)$, then plotting the ordered-eigenvalues from the largest to smallest, we would expect to see a large dip at the (r_k+1)th position. If we do not see such a dip, then we can generate another sample $S_{\text{-}k,m}$ and try again. Obtaining a sample with $d_{\text{-}k}/s_{\text{-}k,m} = O(1)$ should not take long. See the section below.

3.4. *How many samples do we need.* In most applications with $d_k = O(T)$ for each $k \in [K]$, if the ratio $d_{\text{-}k}/s_{\text{-}k,\max} = O(1)$, then Assumption (L2) in Section 3.2.3 is automatically satisfied, and the rate of convergence in (3.12) in Theorem 3.1 becomes $d_k^{-\alpha_{k,1}}$ when we choose $z_k = 1$ there. One way to achieve this is to have $s_{k,max} \asymp d_k$ for each $k \in [K]$.

Consider the scenario where for each $k\in[K]$, $r_k = 1$ and $\mathbf{A}_k$ contains d_k i.i.d. standard normal random variables, with $\mathbf{A}_i$ independent of $\mathbf{A}_j$ for $i \neq j$. For each $S_{k,m} \subseteq [d_k]$ with $m \in [M_{k,0}]$, we want to choose the $S_{k,m}$ such that $\mathbf{A}_k^{\mathrm{T}}\mathbf{1}_{d_k,S_{k,m}}$ is the largest, and that $s_{k,\max} = (\max_{m\in[M_{k,0}]}\mathbf{A}_k^{\mathrm{T}}\mathbf{1}_{d_k,S_{k,m}})^2 \asymp_P d_k$. Now for each $m\in[M_{k,0}]$,
$$z_{k,m} := \frac{\mathbf{A}_k^{\mathrm{T}}\mathbf{1}_{d_k,S_{k,m}}}{\lfloor d_k/2\rfloor^{1/2}} \sim N(0,1), \ \text{ and } \operatorname{corr}(z_{k,m_1}, z_{k,m_2}) = \frac{|S_{k,m_1}\cap S_{k,m_2}|}{\lfloor d_k/2\rfloor},$$
if we are choosing $|S_{k,m}| = \lfloor d_k/2\rfloor$ for each $m\in[M_{k,0}]$. Then by Theorem 3.4 of Hartigan (2014), we have
$$P\Big(\max_{m\in[M_{k,0}]} z_{k,m} \ge \sigma(N + L_\alpha - \frac{1}{2}\log(N+L_\alpha))\Big) \ge 1-2\alpha, \ \text{ where}$$
$$N := \log(M_{k,0}^2/2\pi),\ L_\alpha := -2\log(-\log(\alpha)),\ \sigma := \min_{i\in[M_{k,0}]}\operatorname{var}(z_{k,i} - E(z_{k,i}|z_{k,1},\ldots,z_{k,i-1})),$$
as long as $N + L_\alpha \ge 6$. With $\alpha = 0.025$, then $N+L_\alpha\ge 6$ implies $M_{k,0}\ge 186$, and with this we have
$$P(\max_{m\in[M_{k,0}]} z_{k,m} \ge 5.1\sigma) \ge 0.95,$$

meaning that $s_{k,\max} = (\max_{m\in[M_{k,0}]} \mathbf{A}_k^{\mathrm{T}}\mathbf{1}_{d_k,S_{k,m}})^2$ has order d_k with over 95% probability. Hence if $K=2$, when we are estimating $\mathbf{A}_1$ and to sample fibres from $\mathrm{mat}_1(\mathcal{X}_t)$ using $S_{\text{-}1,\max} = S_{2,\max}$, we have when $M_0 = M_{2,0} \geq 186$ that over 95% probability we can have $s_{\text{-}1,\max} = s_{2,\max} \asymp d_2 = d_{\text{-}1}$.

The value of $M_{k,0}$ in practice to achieve $s_{k,\max} \asymp d_k$ should be smaller than 186 since the constant 5.1σ above can be made smaller. In fact, in practice, we find that around $M_{k,0} = 15$ does a perfect job in all our simulation settings in securing $s_{k,\max} \asymp d_k$. It means that with $K=3$, say we are estimating $\mathbf{A}_2$, then $M_0 = M_{1,0}M_{3,0} = 225$ works fine for securing $s_{\text{-}2,\max} \asymp d_1 d_3 = d_{\text{-}2}$. Indeed in all simulation settings, we use $M_0 = 200$ for $K=2$ or 3 and get very good performance overall.

We do not suggest explicit tuning of M_0, as our pre-averaging estimator is an initial estimator for feeding our iterative projection procedure. Simulation experiments in Section 1.1 in our supplement has clearly shown that the practical performance of our iterative projection estimator remains at a good constant level no matter the initial M_0 we use.

3.5. *Theoretical results for the pre-averaging estimator.* In Section 3.3, we choose $S_{\text{-}k,\max}$ for summing the columns of $\mathrm{mat}_k(\mathcal{X}_t)$. To create stabler estimators, we can construct M different sets $S^{(m)}_{\text{-}k,\max} \subseteq [d_{\text{-}k}]$, $m \in [M]$ (we set $M=5$ in all our simulations), by choosing the best M from the M_0 samples in the procedure laid out in Section 3.3, and form $\widetilde{\mathbf{X}}_{k,1}, \ldots, \widetilde{\mathbf{X}}_{k,M}$, where each $\widetilde{\mathbf{X}}_{k,i}$ is defined in (3.10). Then define

$$\widehat{\Sigma}_{\widetilde{\mathbf{x}}_k,agg} := \frac{1}{M}\sum_{m=1}^{M} \frac{\widetilde{\mathbf{X}}_{k,m}^{\mathrm{T}}\left(\mathbf{I}_T - \frac{1}{T}\mathbf{1}_T\mathbf{1}_T^{\mathrm{T}}\right)\widetilde{\mathbf{X}}_{k,m}}{T}. \tag{3.11}$$

The pre-averaging estimator $\widehat{\mathbf{Q}}_{k,pre,(z_k)}$ is defined as the z_k eigenvectors corresponding to the z_k largest eigenvalues of $\widehat{\Sigma}_{\widetilde{\mathbf{x}}_k,agg}$, with the constraint $\widehat{\mathbf{Q}}^{\mathrm{T}}_{k,pre,(z_k)}\widehat{\mathbf{Q}}_{k,pre,(z_k)} = \mathbf{I}_{z_k}$, for any $z_k \leq r_k$. The theoretical properties of $\widehat{\mathbf{Q}}_{k,pre,(z_k)}$ can be summarized in the following theorem.

THEOREM 3.1. *Let Assumption (E1), (E2), (F1), (L1), (L2), (R1), (R2) be satisfied for all M chosen random samples for constructing $\widehat{\Sigma}_{\widetilde{\mathbf{x}}_k,agg}$, and $r_{e,k} = O(d_k)$. Then*

(3.12)

$$\left\|\widehat{\mathbf{Q}}_{k,pre,(z_k)} - \mathbf{Q}_k\mathbf{H}_k\right\|^2 = O_p\left(d_k^{-2\alpha_{k,z_k}} c_{k,\max}\right), \quad \textit{where}$$

$$c_{k,\max} := \min\left\{1 + \frac{d_k}{T}, \frac{r_k d_k}{T}\right\}\frac{d_{\text{-}k}}{s_{\text{-}k,\max}} + d_k^{\alpha_{k,1}}\left(1 + \frac{d_k^2}{T^2}\right)\frac{d_{\text{-}k}^2}{s_{\text{-}k,\max}^2},$$

$$\mathbf{H}_k := T^{-1}\mathbf{D}_k^{1/2}\frac{1}{M}\sum_{m=1}^{M}\left[\widetilde{\mathbf{F}}_{k,m}\left(\mathbf{I}_T - \frac{1}{T}\mathbf{1}_T\mathbf{1}_T^{\mathrm{T}}\right)\widetilde{\mathbf{F}}^{\mathrm{T}}_{k,m}\right]\mathbf{A}_k^{\mathrm{T}}\widehat{\mathbf{Q}}_{k,pre,(z_k)}\widetilde{\mathbf{V}}_k^{-1}, \quad \textit{with}$$

$$\widetilde{\mathbf{F}}_{k,m} := (\mathrm{mat}_k(\mathcal{F}_{1,\text{-}k})\mathbf{1}_{d_{\text{-}k},S^{(m)}_{\text{-}k,\max}}, \ldots, \mathrm{mat}_k(\mathcal{F}_{T,\text{-}k})\mathbf{1}_{d_{\text{-}k},S^{(m)}_{\text{-}k,\max}}),$$

rank($\mathbf{H}_k$) $= z_k$, and $\widetilde{\mathbf{V}}_k$ is diagonal, containing the z_k eigenvalues (in decreasing order) of $\widehat{\Sigma}_{\widetilde{\mathbf{x}}_k,agg}$. Moreover, further assuming (L1'), there exists $\widehat{\mathbf{U}}_{k,pre,(z_k)}$ with $\widehat{\mathbf{U}}^{\mathrm{T}}_{k,pre,(z_k)}\widehat{\mathbf{U}}_{k,pre,(z_k)} = \mathbf{I}_{z_k}$ such that $\widehat{\mathbf{Q}}_{k,pre,(z_k)} = \widehat{\mathbf{U}}_{k,pre,(z_k)}\mathbf{P}_{k,pre,(z_k)}$ with $\mathbf{P}_{k,pre,(z_k)}$ being an orthogonal matrix, so that

$$\left\|\widehat{\mathbf{U}}_{k,pre,(z_k)} - \mathbf{U}_{k,(z_k)}\right\|^2 = O_p\left(d_k^{-2\alpha_{k,z_k}}\left[d_k^{2\alpha_{k,1}}\frac{r_k}{T} + c_{k,\max}\right]\right). \tag{3.13}$$

The matrix $\mathbf{U}_{k,(z_k)}$ is defined to be the matrix consisting of the first z_k columns of $\mathbf{U}_k$.

The meanings for (3.12) and (3.13) are different. When $z_k < r_k$, (3.12) suggests that the estimated directions $\widehat{\mathbf{Q}}_{k,pre,(z_k)}$ will lie in the subspace spanned by the columns of $\mathbf{Q}_k$ (or $\mathbf{U}_k$), but it may not be "close" to the directions corresponding to the strongest z_k factors. However, with (3.13), we can conclude that $\widehat{\mathbf{U}}_{k,pre,(z_k)}$ will be "close" to the directions which correspond to the strongest z_k factors. As a compromise, (3.13) involves an extra rate $d_k^{2(\alpha_{k,1}-\alpha_{k,z_k})} r_k T^{-1}$ as compared to (3.12). Such a difference is especially notable when we set $z_k = 1$ and perform the iterative projection in Section 4.

Remark: Suppose in (L2), the ratio $d_{\text{-}k}/s_{\text{-}k,max}$ is of order $d_{\text{-}k}^{-1}$, which can be achieved if, for instance, there exists a dense column in $\mathbf{A}_j$ (i.e., pervasive factor) having majority of elements of the same sign for each $j \in [K]$. Suppose further that the r_k's and K are constants, with $d_k \asymp T$ for each $k \in [K]$. The results from Theorem 3.1 implies that the projection matrix $\widehat{\mathbf{P}}_{k,pre} := \widehat{\mathbf{Q}}_{k,pre,(r_k)} \widehat{\mathbf{Q}}^{\mathsf{T}}_{k,pre,(r_k)}$ has error rate

$$\begin{aligned} \big\|\widehat{\mathbf{P}}_{k,pre} - \mathbf{Q}_k(\mathbf{Q}_k^{\mathsf{T}}\mathbf{Q}_k)^{-1}\mathbf{Q}_k^{\mathsf{T}}\big\| &= \big\|\widehat{\mathbf{P}}_{k,pre} - \mathbf{U}_k\mathbf{U}_k^{\mathsf{T}}\big\| \\ &= O_P(d_k^{-\alpha_{k,r_k}}(d_{\text{-}k}^{-1/2} + d_k^{\alpha_{k,1}/2} d_{\text{-}k}^{-1})). \end{aligned} \tag{3.14}$$

This can be compared to the rates in Chen, Yang and Zhang (2022), which need the errors to be sub-Gaussian (compared to our Assumption (R1) where only bounded fourth moments is needed). While their σ^2 can be considered constant, their λ is such that $\lambda \asymp \prod_{k=1}^K d_k^{\alpha_{k,1}}$. The TIPUP procedure has rate (in our notations, using equation (47) in Chen, Yang and Zhang (2022), which has a faster rate of convergence than TOPUP)

$$\big\|\widehat{\mathbf{P}}_k - \mathbf{U}_k\mathbf{U}_k^{\mathsf{T}}\big\| = O_P\left(\frac{d_k^{1/2}}{T^{1/2}\prod_{k=1}^K d_k^{\alpha_{k,1}/2}} + \frac{d^{1/2}}{T^{1/2}\prod_{k=1}^K d_k^{\alpha_{k,1}}}\right). \tag{3.15}$$

When all factors are strong, i.e., $\alpha_{k,j} = 1$, the rate in (3.14) is faster than that in (3.15). When $\alpha_{k,1} = 1$ and $\alpha_{k,r_k} = 0.5$, i.e., the strongest factor is pervasive but the weakest factor is quite weak, then the two rates will be the same.

The rate in (3.14) can also be compared to Theorem 1 of Chen and Fan (2021) when $K = 2$, which under the same conditions laid out at the start of the remark, implies

$$\big\|\widehat{\mathbf{P}}_k - \mathbf{U}_k\mathbf{U}_k^{\mathsf{T}}\big\| = O_P(d_k^{-1/2}). \tag{3.16}$$

Our rate in (3.14) is $d_k^{-3/2}$ when all factors are strong, and is d_k^{-1} when $\alpha_{k,1} = 1$ and $\alpha_{k,r_k} = 0.5$. Both rates are faster than $d_k^{-1/2}$ in (3.16).

Indeed, the better performance of the iterative projection estimator, which uses the pre-averaging estimator as an initial estimator, is reflected in the empirical results in Section 6.

3.6. *A discussion on optimality.* Our pre-averaging estimator achieves a minimax optimal rate under certain scenarios over a certain localized set. For simplicity, suppose we only take $M = 1$ in (3.11), and assume the data has mean 0. It means from (3.1) that

$$\widehat{\Sigma}_{\widetilde{\mathbf{x}}_k,agg} = \frac{1}{T}\widetilde{\mathbf{X}}_{k,1}^{\mathsf{T}}\widetilde{\mathbf{X}}_{k,1} = \mathbf{M}^* + \mathbf{H}, \text{ where}$$

$$\begin{aligned} \mathbf{H} := &\frac{1}{T}\sum_{t=1}^T(\mathbf{A}_k\mathbf{F}_t\mathbf{A}_{\text{-}k}^{\mathsf{T}}\mathbf{q}\mathbf{q}^{\mathsf{T}}\mathbf{E}_t^{\mathsf{T}} + \mathbf{E}_t\mathbf{q}\mathbf{q}^{\mathsf{T}}\mathbf{A}_{\text{-}k}\mathbf{F}_t^{\mathsf{T}}\mathbf{A}_k^{\mathsf{T}}) \\ &+ \frac{1}{T}\sum_{t=1}^T(\mathbf{E}_t\mathbf{q}\mathbf{q}^{\mathsf{T}}\mathbf{E}_t^{\mathsf{T}} - E[\text{diag}(\mathbf{E}_t\mathbf{q}\mathbf{q}^{\mathsf{T}}\mathbf{E}_t^{\mathsf{T}})]), \end{aligned}$$

$$\mathbf{M}^* := \frac{1}{T}\sum_{t=1}^{T}\mathbf{A}_k\mathbf{F}_t\mathbf{A}_{\text{-}k}^{\mathsf{T}}\mathbf{q}\mathbf{q}^{\mathsf{T}}\mathbf{A}_{\text{-}k}\mathbf{F}_t^{\mathsf{T}}\mathbf{A}_k^{\mathsf{T}} + \frac{1}{T}\sum_{t=1}^{T}E[\mathrm{diag}(\mathbf{E}_t\mathbf{q}\mathbf{q}^{\mathsf{T}}\mathbf{E}_t^{\mathsf{T}})],$$

with $\mathbf{F}_t := \mathrm{mat}_k(\mathcal{F}_t)$, $\mathbf{E}_t := \mathrm{mat}_k(\mathcal{E}_t)$ and $\mathbf{q} := \mathbf{1}_{d_{\text{-}k},S_{\text{-}k,\max}}/\|\mathbf{1}_{d_{\text{-}k},S_{\text{-}k,\max}}\|$ (normalizing it does not affect the eigenvectors). Assume also $\mathbf{E}_t$ has only i.i.d. entries with finite 4th order moments (i.e., $\mathcal{E}_t = \boldsymbol{\epsilon}_t$ in (E1), each element having the same finite variance), so that $E(\mathbf{H}) = \mathbf{0}$, and $T^{-1}\sum_{t=1}^{T}E[\mathrm{diag}(\mathbf{E}_t\mathbf{q}\mathbf{q}^{\mathsf{T}}\mathbf{E}_t^{\mathsf{T}})] = \sigma_\epsilon^2\mathbf{I}_{d_k}$, where $\sigma_\epsilon^2 = \mathrm{var}((\mathbf{E}_t)_{ij})$.

Let λ_j^* be the j-th largest eigenvalue of $\mathbf{M}^*$. The set of eigenvectors for $\mathbf{M}^*$ now coincides with the columns in $\mathbf{U}_k$ defined in (3.9), and we write $\mathbf{u}_j^*$ to be the j-th column of $\mathbf{U}_k$. Following equation (20c) in Cheng, Wei and Chen (2021), define

$$\mathcal{M}(\mathbf{M}^*) := \Big\{ A \in \mathbb{R}^{d_k\times d_k} \text{ symmetric} \mid \mathrm{rank}(A) = r_k,$$
$$\lambda_i(A) = \lambda_i^* \;\; (1 \le i \le r_k),\; \|\mathbf{u}_j(A) - \mathbf{u}_j^*\| \le \frac{c\sigma_{\min}\sqrt{d_k}}{|\lambda_j^*|}\Big\},$$

where $u_j(A)$ is the eigenvector corresponding to the j-th largest eigenvalue of A, and $\sigma_{\min}^2$ is the smallest value amongst of the variance of the elements of $\mathbf{H}$.

We can easily show that, as $T \to \infty$,

$$\lambda_j^* \asymp_P d_k^{\alpha_{k,j}}\mathbf{q}^{\mathsf{T}}\mathbf{A}_{\text{-}k}\mathbf{A}_{\text{-}k}^{\mathsf{T}}\mathbf{q} \asymp \frac{d_k^{\alpha_{k,j}}s_{\text{-}k,\max}}{d_{\text{-}k}},\;\; j \in [r_k];\;\; \sigma_{\min} \asymp \sqrt{\frac{\mathbf{q}^{\mathsf{T}}\mathbf{A}_{\text{-}k}\mathbf{A}_{\text{-}k}^{\mathsf{T}}\mathbf{q}}{T}} \asymp \sqrt{\frac{s_{\text{-}k,\max}}{Td_{\text{-}k}}}.$$

Then the conditions in Theorem 3 of Cheng, Wei and Chen (2021) are satisfied, except that the elements of $\mathbf{H}$ are at most asymptotically normal as $T \to \infty$, and are dependent in general. The conclusion of the theorem is that for $j \in [r_k]$,

$$\inf_{\widehat{\mathbf{u}}_j}\sup_{A\in\mathcal{M}(\mathbf{M}^*)} E\|\widehat{\mathbf{u}}_j - \mathbf{u}_j(A)\| \ge \frac{C\sigma_{\min}\sqrt{d_k}}{|\lambda_j^*|} \asymp \frac{1}{d_k^{\alpha_{k,j}}}\sqrt{\frac{d_k}{T\mathbf{q}^{\mathsf{T}}\mathbf{A}_{\text{-}k}\mathbf{A}_{\text{-}k}^{\mathsf{T}}\mathbf{q}}} \asymp \frac{1}{d_k^{\alpha_{k,j}}}\sqrt{\frac{d}{Ts_{\text{-}k,\max}}}.$$

Similar to the remark at the end of Section 3.5, suppose $s_{\text{-}k,\max} \succeq d_{\text{-}k}d_k^{\alpha_{k,j}}$, which can be achieved if there exists a column in $\mathbf{A}_\ell$ having "enough" elements of the same sign for each $\ell \in [K]$ (if all are of the same sign, then $s_{\text{-}k,\max} \asymp d_{\text{-}k}^2$, which can be much larger than $d_{\text{-}k}d_k^{\alpha_{k,j}}$). Suppose also r_k and K are constants, and $d_\ell \asymp T$ for each $\ell \in [K]$. Then the minimax rate above is $d_k^{-\alpha_{k,j}}(d_{\text{-}k}/s_{\text{-}k,\max})^{1/2}$ for $j \in [r_k]$, which coincides with the rate from Theorem 3.1 for the pre-averaging estimator when $z_k = j \le r_k$:

$$\|\widehat{\mathbf{Q}}_{k,pre,(j)} - \mathbf{Q}_k\mathbf{H}_k\| = O_P(d_k^{-\alpha_{k,j}}(d_{\text{-}k}/s_{\text{-}k,\max})^{1/2}) \;\text{ for } j \in [r_k],\; \text{implying}$$
$$\|\widehat{\mathbf{P}}_{k,pre} - \mathbf{U}_k\mathbf{U}_k^{\mathsf{T}}\| = O_P(d_k^{-\alpha_{k,r_k}}(d_{\text{-}k}/s_{\text{-}k,\max})^{1/2}),$$

where $\widehat{\mathbf{P}}_{k,pre} := \widehat{\mathbf{Q}}_{k,pre,(r_k)}\widehat{\mathbf{Q}}_{k,pre,(r_k)}^{\mathsf{T}}$. Define $\mathbf{U}(A) := (\mathbf{u}_1(A),\ldots,\mathbf{u}_{r_k}(A))$, then

$$\sup_{A\in\mathcal{M}(\mathbf{M}^*)}\|\widehat{\mathbf{P}}_{k,pre} - \mathbf{U}(A)\mathbf{U}(A)^{\mathsf{T}}\| \le \|\widehat{\mathbf{P}}_{k,pre} - \mathbf{U}_k\mathbf{U}_k^{\mathsf{T}}\| + \sup_{A\in\mathcal{M}(\mathbf{M}^*)} 2\|\mathbf{U}(A) - \mathbf{U}_k\|$$
$$= O_P(d_k^{-\alpha_{k,r_k}}(d_{\text{-}k}/s_{\text{-}k,\max})^{1/2}).$$

4. Re-estimation by Projection. While Yu et al. (2022), He et al. (2023a) and He, Li and Trapani (2022) all deal with projection estimation of a factor loading matrix in the case of $K = 2$ or a general K, they all assume that all factors are pervasive. And in practice, they need to know the number of factors r_k in $\mathbf{A}_k$ for each $k \in [K]$ first in order to estimate a projection matrix $\mathbf{B}_k$ of size $d_{\text{-}k} \times r_{\text{-}k}$, where $r_{\text{-}k} := r/r_k$ with $r = r_1\cdots r_K$.

In contrast, our projection method to be presented here does not need the estimation of each r_k first, since we are projecting to one direction only: the direction of the *strongest* factors, iteratively. Setting $z_k = 1$, the pre-averaging vector $\widehat{\mathbf{Q}}_{k,pre,(1)}$ is indeed asymptotically pointing to the direction of the strongest factors (see (3.13) in Theorem 3.1).

Projecting to the direction of the strongest factors is needed in our setting since there are weak factors. Their estimators have worse rate of convergence and estimation performance than pervasive ones. Using these worse estimated directions for projections will deteriorate the performance of the projection estimators. In Section 6, we demonstrated that under the presence of weak factors, our method provides the best performance of factor loading matrix estimation compared to all other state-of-the-art methods, including the projection estimation suggested by these three papers.

In (3.2), we demean the data first and change the projection direction to $\mathbf{q}_{\text{-}k}$, where

$$\mathbf{q}_{\text{-}k} := \mathbf{q}_K \otimes \cdots \otimes \mathbf{q}_{k+1} \otimes \mathbf{q}_{k-1} \otimes \cdots \otimes \mathbf{q}_1, \;\text{ with }\; \mathbf{q}_k := \mathbf{A}_k\mathbf{c}_k,\; k \in [K],$$

for some non-zero constant vectors $\mathbf{c}_k$. Then defining $\mathbf{c}_{\text{-}k} := \mathbf{c}_K \otimes \cdots \mathbf{c}_{k+1} \otimes \mathbf{c}_{k-1} \otimes \cdots \otimes \mathbf{c}_1$, we have $\mathbf{q}_{\text{-}k} = \mathbf{A}_{\text{-}k}\mathbf{c}_{\text{-}k}$, and we can construct the new projected data as

$$\begin{aligned}\mathbf{y}_t^{(k)} &:= \text{mat}_k(\mathcal{X}_t - \bar{\mathcal{X}})\mathbf{q}_{\text{-}k} \\ &= \mathbf{A}_k \text{mat}_k(\mathcal{F}_t - \bar{\mathcal{F}})\mathbf{A}_{\text{-}k}^{\mathrm{T}}\mathbf{A}_{\text{-}k}\mathbf{c}_{\text{-}k} + \text{mat}_k(\mathcal{E}_t - \bar{\mathcal{E}})\mathbf{q}_{\text{-}k}.\end{aligned} \tag{4.1}$$

Depending on the direction $\mathbf{c}_{\text{-}k}$, we can see from above that the signals from the factors are strengthened due to the term $\mathbf{A}_{\text{-}k}^{\mathrm{T}}\mathbf{A}_{\text{-}k}\mathbf{c}_{\text{-}k}$, while the noise level is retained or strengthened, depending on the level of cross-correlations among the noise fibres. The projected data can also be used to estimate a finer projection direction, essentially iterating the projection step. See Theorem 4.1 below and the explanations followed. See simulation results regarding this in Section 6 as well.

4.1. *Refining the projection direction.*

From Theorem 3.1, setting $z_k = 1$ there, we obtain $\widehat{\mathbf{q}}_{k,pre} := \widehat{\mathbf{Q}}_{k,pre,(1)} = \widehat{\mathbf{U}}_{k,pre,(1)}\mathbf{P}_{k,pre,(1)} = \pm\widehat{\mathbf{U}}_{k,pre,(1)}$ (WLOG we take the plus sign in the presentations hereafter). For each $k \in [K]$, we create the projected data $\mathbf{y}_t^{(k)}$ as in (4.1), using

$$\mathbf{q}_{\text{-}k} = \widehat{\mathbf{q}}_{\text{-}k,pre} := \widehat{\mathbf{q}}_{K,pre} \otimes \cdots \otimes \widehat{\mathbf{q}}_{k+1,pre} \otimes \widehat{\mathbf{q}}_{k-1,pre} \otimes \cdots \otimes \widehat{\mathbf{q}}_{1,pre}. \tag{4.2}$$

Then we define $\check{\mathbf{q}}_k^{(1)}$ to be the eigenvector corresponding to the largest eigenvalue of the matrix

$$\widetilde{\mathbf{\Sigma}}_y^{(k)} := T^{-1}\sum_{t=1}^T \mathbf{y}_t^{(k)}\mathbf{y}_t^{(k)\mathrm{T}}.$$

The superscript (1) in $\check{\mathbf{q}}_k^{(1)}$ signals that this is the first iterated estimator for $\mathbf{U}_{k,(1)}$. We can iterate this process to obtain refinement of projection direction. More formally, we introduce the following algorithm.

Algorithm for Iterative Projection Direction Refinement

1. Initialize $\check{\mathbf{q}}_k^{(0)} = \widehat{\mathbf{q}}_{k,pre}$ for each $k \in [K]$.
2. For $i \geq 1$, at the i-th step, create projected data $\mathbf{y}_{t,i}^{(k)} := \text{mat}_k(\mathcal{X}_t - \bar{\mathcal{X}})\check{\mathbf{q}}_k^{(i-1)}$ for each $k \in [K]$.
3. For each $k \in [K]$, define $\check{\mathbf{q}}_k^{(i)}$ the eigenvector corresponding to the largest eigenvalue of

$$\widetilde{\mathbf{\Sigma}}_{y,i}^{(k)} := T^{-1}\sum_{t=1}^T \mathbf{y}_{t,i}^{(k)}\mathbf{y}_{t,i}^{(k)\mathrm{T}}. \tag{4.3}$$

4. Replace i by $i+1$. Go back to step 2. Stop until after the procedure has been repeated for a fixed number of times.

We present a further assumption needed before presenting Theorem 4.1.

(RE1) *For a positive integer N, let $\mathcal{A}_{f,T} \in \mathbb{R}^{(N+1)T\times T}$ be defined as $\mathcal{A}_{f,T} := (\mathbf{a}_{f,1}, \dots, \mathbf{a}_{f,T})$, where*

$$\mathbf{a}_{f,t} := (\mathbf{0}_{t-1}^{\mathrm{T}}, a_{f,NT}, a_{f,NT-1}, \dots, a_{f,0}, \mathbf{0}_{T-t}^{\mathrm{T}})^{\mathrm{T}}, \quad t \in [T],$$

with $\mathbf{0}_j$ being a column vector of j zeros and the $a_{f,q}$'s are from Assumption (F1). Define $\mathcal{A}_{e,T}$ and $\mathcal{A}_{\epsilon,T}$ similarly using coefficients from $\{a_{e,q}\}$ and $\{a_{\epsilon,q}\}$ respectively from Assumption (E2). Then we assume that (with $\mathcal{A}$ can be either $\mathcal{A}_{f,T}, \mathcal{A}_{e,T}$ or $\mathcal{A}_{\epsilon,T}$) $\|\mathcal{A}\|$ is uniformly bounded above, and

$$\frac{1}{T}\mathrm{tr}(\mathcal{A}^{\mathrm{T}}\mathcal{A}) = 1 - o(T^{-2}d^{-2}),$$

$$\frac{1}{T}\mathrm{tr}(\mathcal{A}^{\mathrm{T}}\mathcal{A})^2 \to a_1, \quad \frac{1}{T^2}\mathbf{1}_T^{\mathrm{T}}(\mathcal{A}^{\mathrm{T}}\mathcal{A})^2\mathbf{1}_T \to a_2, \quad \frac{1}{T^{3/2}}\mathbf{1}_T^{\mathrm{T}}\mathcal{A}^{\mathrm{T}}\mathcal{A}\mathbf{1}_T \to a_3,$$

where $\mathbf{1}_T$ is a column vector of T ones, and the constants a_1, a_2 and a_3 can be different for $\mathcal{A} = \mathcal{A}_{f,T}, \mathcal{A}_{e,T}$ and $\mathcal{A}_{\epsilon,T}$ respectively.

Consider a truncated linear process $\{y_t\}_{t\in[T]}$, and the original process $\{\widetilde{y}_t\}_{t\in[T]}$,

$$\widetilde{y}_t = \sum_{q\geq 0} a_q z_{t-q}, \quad y_t = \sum_{q=0}^{NT} a_q z_{t-q}, \text{ with } \mathrm{var}(\widetilde{y}_t) = 1,$$

where $\{z_t\}$ is a sequence of i.i.d. random variables. Construct the matrix $\mathcal{A}$ using $\{a_q\}$ similar to those in Assumption (RE1). Then $\mathcal{A}^{\mathrm{T}}\mathcal{A}$ contains the variance of $\{y_t\}$ on the diagonal, and lag-k autocovariance on the k-th off-diagonal. The rates in (RE1) are then controlling how fast the a_q's are going to 0, and how much serial dependence between the y_t's are allowed. In particular, general linear processes with absolutely summable autocovariance sequence, short range dependent processes like ARMA models, satisfy the assumption.

THEOREM 4.1. *Let all the assumptions in Theorem 3.1 be satisfied, together with (RE1). Let $g_s := \prod_{j=1}^K d_j^{\alpha_{j,1}}$, $r_e := \prod_{k=1}^K r_{e,k}$. Assume further that for each $k \in [K]$, $r = O(dg_s^{-1})$, $r_e = o(T)$, $d_k = O(g_s) = (r_e + \sqrt{T/r})$. Then*

$$\|\check{\mathbf{q}}_k^{(1)} - \mathbf{U}_{k,(1)}\| = O_P\bigg\{\sqrt{\frac{r}{T}} + g_s^{-1/2} b_k \sqrt{\frac{rd}{T}}\bigg\}, \text{ (assumed } o_P(1)) \text{ where}$$

$$b_k = K\sqrt{\frac{r_{\max}}{T}} + \sum_{j=1;j\neq k}^{K} d_j^{-\alpha_{j,1}} c_{j,max}^{1/2} = o(1).$$

Furthermore, if $rdg_s^{-1} = o(T)$, then for an integer $m \geq 1$,

$$\|\check{\mathbf{q}}_k^{(m+1)} - \mathbf{U}_{k,(1)}\| = O_P\bigg\{\sqrt{\frac{r}{T}} + g_s^{-1/2}\|\check{\mathbf{q}}_{\text{-}k}^{(m)} - \mathbf{U}_{\text{-}k,(1)}\|\sqrt{\frac{rd}{T}}\bigg\} = o_P(1),$$

and the Algorithm for Iterative Projection Direction Refinement will produce, after a certain number of iterations (say m),

$$\|\check{\mathbf{q}}_k^{(m)} - \mathbf{U}_{k,(1)}\| = O_P\bigg(\sqrt{\frac{r}{T}}\bigg).$$

To put the above results into perspective, assume a very common scenario that $d_1 \asymp \cdots \asymp d_K \asymp T$ (this is especially true in economic applications where T is small), with K and each r_k being constants for $k \in [K]$. We first note that if all r_k factors in $\mathbf{A}_k$ are pervasive, i.e., $\alpha_{k,j} = 1$ for all $j \in [r_k]$, then $g_s = d$, and hence $\big\|\check{\mathbf{q}}_k^{(1)} - \mathbf{U}_{k,(1)}\big\| = O_P(T^{-1/2})$, and any refinements will retain the same rate. Even if $\alpha_{k,1} < 1$ (i.e., the strongest factor corresponding to $\mathbf{A}_k$ is not pervasive), $\big\|\check{\mathbf{q}}_k^{(1)} - \mathbf{U}_{k,(1)}\big\|$ can still be $O_P(T^{-1/2})$, as long as $b_k^2 d/g_s = O(1)$, equivalent to $\alpha_{k,1} \geq 1/2$. The case of $\alpha_{k,1} = 1/2$ presents a significantly weak strongest factor corresponding to $\mathbf{A}_k$, and without the help of projection and strong factors from other modes' factor loading spaces, the typical rate for estimating such a weak factor would be $d_k^{-1/4}$ which is much worse than $T^{-1/2}$.

To have an idea on the value of m, from the last part of the proof of Theorem 4.1, we need

$$b_k \left(\sqrt{\frac{rd}{Tg_s}}\right)^m = O\left(\sqrt{\frac{r}{T}}\right).$$

Suppose $d_k \asymp T$, r_k is a constant and $d_{\text{-}k}/s_{\text{-}k,\max} \asymp 1$ (see Section 3.4 on how to achieve this). Then $b_k \asymp d_k^{-\alpha_{k,1}/2}$, and hence

$$m \geq \frac{\text{constant} + \alpha_{k,1}\log(d_k) - \log(T)}{\log(\frac{rd}{Tg_s})}.$$

Further, if $\alpha_{k,1} = 0.5$ (a very weak factor), and $d/g_s \asymp T^{0.95}$ (recall that we assume $rdg_s^{-1} = o(T)$), then as $T, d_k \to \infty$, we have $m \geq 10$. This is already quite extreme since $d/g_s \asymp T^{0.95}$ means that the strongest factors of some other $\mathbf{A}_k$'s are also weak. The fact that we are using $m = 30$ in our simulations in Section 6 throughout made sure that the rate $\sqrt{r/T}$ is reached, and we do not recommend users increase m further for saving computational time.

The fixed rate $O_P(\sqrt{r/T})$ in Theorem 4.1 comes from the fact that we need to distinguish the direction of the strongest factors from all other directions of weaker factors in order to find the "best" projection direction. In the case of studying the whole $\mathbf{U}_k$, we in fact may get a better rate of convergence even in the presence of weak factors.

THEOREM 4.2. *Let all the assumptions in Theorem 4.1 be satisfied. Suppose we know the value of r_k, and perform an eigenanalysis on $\widetilde{\boldsymbol{\Sigma}}_{y,m+1}^{(k)}$ in (4.3) which utilized the projection direction $\check{\mathbf{q}}_k^{(m)}$ in Theorem 4.1, obtaining r_k eigenvectors as an estimator of the factor loading space of $\mathbf{A}_k$.*

Then there exists $\check{\mathbf{U}}_k \in \mathbb{R}^{d_k \times r_k}$ with $\check{\mathbf{U}}_k^{\mathrm{T}}\check{\mathbf{U}}_k = \mathbf{I}_{r_k}$ such that the r_k eigenvectors obtained above is $\check{\mathbf{U}}_k$ multiplied with some orthogonal matrix, with

$$\big\|\check{\mathbf{U}}_k - \mathbf{U}_k\big\| = O_P\left\{d_k^{\alpha_{k,1}-\alpha_{k,r_k}}\left[g_s^{-1} + \sqrt{\frac{r}{Tg_s}}\left(r_e^{1/2} + d_k^{1/2} + \sqrt{\frac{rd}{T}}\right)\right]\right\}, \ (\textit{assumed } o_P(1)).$$

Consider $d_1 \asymp \cdots \asymp d_K \asymp T$, with K and r_k being constants for $k \in [K]$. If all factors for $\mathbf{A}_k$ are pervasive, i.e., $\alpha_{k,j} = 1$ for all $j \in [r_k]$, then we have $\big\|\check{\mathbf{U}}_k - \mathbf{U}_k\big\| = O_P(T^{-1})$. When $K = 2$, this has the same rate as the average Frobenius error norm of the estimators of $\mathbf{A}_1$ and $\mathbf{A}_2$ in Theorem 3.1 and Theorem 4.1 of He et al. (2023a). This is also consistent with the rate in Corollary 3.1 of He, Li and Trapani (2022), Theorem 3.1 and 3.2 of He et al. (2022), and Theorem 3.1 of Yu et al. (2022) under the same scenario.

The above rate can be greatly improved if the term $\sqrt{rd/T}$ can be removed. It is there because the estimated projection direction is correlated with the data in general. If we have independent noise tensor $\{\mathcal{E}_t\}$ (e.g., the setting in Chen, Yang and Zhang (2022)) we can split the data into half, and using only one half of it for projection direction estimation while the other half is for re-estimation only. Then the estimated projection direction will be independent of the re-estimation data, and hence the final rate indeed will be rid of this term. When all the factors are strong, this improved rate will be the same as the one for TIPUP in equation (47) of Chen, Yang and Zhang (2022). We do not pursue this since our paper focuses on time series data with serial correlation in the noise. Moreover, the empirical performance of our projection method is very good already.

5. Core Tensor Rank Estimation Using Projected Data. With the projected data and the associated covariance matrix $\widetilde{\mathbf{\Sigma}}_{y,m+1}^{(k)}$ defined in (4.3), define

$$\widetilde{\mathbf{R}}_{y,m+1}^{(k)} := \text{diag}^{-1/2}(\widetilde{\mathbf{\Sigma}}_{y,m+1}^{(k)})\widetilde{\mathbf{\Sigma}}_{y,m+1}^{(k)}\text{diag}^{-1/2}(\widetilde{\mathbf{\Sigma}}_{y,m+1}^{(k)}),\ k \in [K]. \tag{5.1}$$

Our estimator for r_k for each $k \in [K]$ is then defined to be

$$\widehat{r}_k := \max\{j : \lambda_j(\widetilde{\mathbf{R}}_{y,m+1}^{(k)}) > 1 + \eta_T,\ j \in [d_k]\}, \tag{5.2}$$

where $\eta_T \to 0$ as $T \to \infty$, and its practical choice will be discussed in Section 5.2. This estimator is inspired by the one in Fan, Guo and Zheng (2022) for independent observations from a vector factor model.

5.1. *Main results.* The following assumption is needed for all the theorems in this section.

(RE2) (Model Parameters) *For each $k \in [K]$, we assume that for each $j \in [d_k]$, the value $\lambda_j(\text{diag}(\mathbf{A}_k\mathbf{A}_k^{\mathrm{T}}))$ is uniformly bounded away from 0 and infinity as $T, d_k \to \infty$. Moreover, $r_k = o(d_k^{1-\alpha_{k,1}+\alpha_{k,r_k}})$.*

Assumption (RE2) ensures that each row of $\mathbf{A}_k$ has at least one non-zero value, meaning that at least one factor drives the dynamics of the corresponding element in $\mathbf{y}_{t,m+1}^{(k)}$. The assumption can be weakened so that the values are vanishing, at the price of more complicated proofs and rates in Theorem 5.2. Define

$$\mathbf{\Sigma}_{y,m+1}^{(k)} := \check{\mathbf{q}}_{\text{-}k}^{(m)\mathrm{T}}\mathbf{A}_{\text{-}k}\mathbf{A}_{\text{-}k}^{\mathrm{T}}\check{\mathbf{q}}_{\text{-}k}^{(m)}\mathbf{A}_k\mathbf{A}_k^{\mathrm{T}} + \sum_{j=1}^{d_{\text{-}k}}(\check{\mathbf{q}}_{\text{-}k}^{(m)})_j^2\mathbf{\Sigma}_{\epsilon,j}^{(k)} + \check{\mathbf{q}}_{\text{-}k}^{(m)\mathrm{T}}\mathbf{A}_{e,\text{-}k}\mathbf{A}_{e,\text{-}k}^{\mathrm{T}}\check{\mathbf{q}}_{\text{-}k}^{(m)}\mathbf{A}_{e,k}\mathbf{A}_{e,k}^{\mathrm{T}}. \tag{5.3}$$

The matrix $\mathbf{\Sigma}_{y,m+1}^{(k)}$ is in fact the expected value of $\widetilde{\mathbf{\Sigma}}_{y,m+1}^{(k)}$ in (4.3), pretending that $\check{\mathbf{q}}_{\text{-}k}^{(m)}$ is a constant vector.

THEOREM 5.1. *Let Assumption (E1), (F1) and (RE2) hold. Define the correlation matrix*

$$\mathbf{R}_{y,m+1}^{(k)} = \text{diag}^{-1/2}(\mathbf{\Sigma}_{y,m+1}^{(k)})\mathbf{\Sigma}_{y,m+1}^{(k)}\text{diag}^{-1/2}(\mathbf{\Sigma}_{y,m+1}^{(k)}),\quad k \in [K].$$

Then for large enough T, d_k, we have in probability $\lambda_j(\mathbf{R}_{y,m+1}^{(k)}) \succeq_P r_k^{-1}d_k^{1-\alpha_{k,1}+\alpha_{k,j}} > 1$ for $j \in [r_k]$, whereas $\lambda_j(\mathbf{R}_{y,m+1}^{(k)}) \leq 1$ for $j = r_k + 1, \ldots, d_k$.

This theorem is in parallel to Theorem 1 of Fan, Guo and Zheng (2022). With this, we can write

$$r_k = \max\{j : \lambda_j(\mathbf{R}_{y,m+1}^{(k)}) > 1,\ j \in [d_k]\}.$$

In light of this, the estimator $\widehat{r}_k$ in (5.2) makes sense. The following theorem shows further that $\widehat{r}_k$ is in fact consistent for r_k for a suitable choice of η_T.

THEOREM 5.2. *Let (RE2) and all the assumptions in Theorem 4.1 hold. Suppose*

$$d_k^{\alpha_{k,1}-\alpha_{k,r_k}}\left(\sqrt{\frac{r(r_e+d_k)}{Tg_s}}+\frac{Kr}{T}\sqrt{\frac{d}{g_s}}\right)=o(1),\quad k\in[K],$$

where g_s is defined in Theorem 4.1. Then as $T, d_k\to\infty$, we have for each $k\in[K]$,

$$\lambda_j(\widetilde{\mathbf{R}}_{y,m+1}^{(k)})=\begin{cases}\succeq_P r_k^{-1}d_k^{1-\alpha_{k,1}+\alpha_{k,j}} & \\ \quad\cdot(1+O_P\{r_kd_k^{2\alpha_{k,1}-\alpha_{k,j}-1}a_T(0)+a_T(\alpha_{k,1})\}), & j\in[r_k];\\ \leq 1+O_P\{b_T\}, & j\in[d_k]/[r_k],\end{cases}$$

where for $0<\delta\leq 1$,

$$a_T(\delta):=\sqrt{\frac{r}{T}}\left[1+d_k^{\delta/2}g_s^{-1/2}\left(r_e^{1/2}+d_k^{1/2}+K\sqrt{\frac{rd}{T}}\right)+d_k^{\delta}g_s^{-1}\frac{K^2r^{1/2}d}{T^{3/2}}\right],$$

$$b_T:=d_k^{\alpha_{k,1}}g_s^{-1}\left\{\sqrt{\frac{(r_e+d_k)d_k}{T}}+\frac{K\sqrt{r(r_e+d_k)d}}{T}+\frac{K^2rd}{T^2}\right\},$$

with $r_kd_k^{2\alpha_{k,1}-\alpha_{k,r_k}-1}a_T(0)$, $a_T(\alpha_{k,1})$ and b_T assumed $o(1)$. Hence $\widehat{r}_k$ in (5.2) is a consistent estimator for r_k if we choose $\eta_T=Cb_T$ for some constant $C>0$.

To gain some insights from the theorem, suppose the strongest factor for each mode-k unfolded matrix is pervasive, i.e., $\alpha_{j,1}=1$ for each $j\in[K]$, and r_k and K are constants with $d_1\asymp\cdots\asymp d_K\asymp T$. Then

$$r_kd_k^{2\alpha_{k,1}-\alpha_{k,j}-1}a_T(0)+a_T(1)\asymp T^{-1/2},\quad b_T=O(T^{1/2}d_{\text{-}k}^{-1}+d_{\text{-}k}^{-1/2}+T^{-1}).$$

This shows that the rate of convergence of b_T is at best $T^{-1/2}$ when $K=2$, and T^{-1} when $K\geq 3$. It means that our search for η_T can be in the form $CT^{-1/2}$ when $K=2$, and CT^{-1} when $K\geq 3$. The extra rate assumptions in the theorem may not be more stringent than those in Theorem 4.1 and (RE2). For instance, if K and each r_k for $k\in[K]$ are constants with $d_1\asymp\cdots\asymp d_K\asymp T$ and all factors are pervasive, then the extra rate assumptions in Theorem 5.2 are satisfied automatically.

5.2. *Practical implementation for core rank estimator.* Since there is only one mode-k unfolding matrix from our data, we propose the following algorithm for Bootstrapping the mode-k fibres to facilitate the search for η_T.

Bootstrapping Algorithm for mode-k tensor fibres and projected data

1. Initialize an integer $B>0$, and independent sequences of i.i.d. Bernoulli random variables $\{\xi_j^{(b)}\}_{j\in[d_{\text{-}k}]}$ for each $b\in[B]$.
2. For each b, create $\mathbf{W}_b\in\mathbb{R}^{d_{\text{-}k}\times d_{\text{-}k}}$, where the i-th column is $\mathbf{0}$ except its j-th zero is replaced by $\xi_i^{(b)}$, with j chosen uniformly from $[d_{\text{-}k}]$.
3. Define new projected data $\mathbf{y}_{t,m+1,b}^{(k)}:=\text{mat}_k(\mathcal{X}_t-\bar{\mathcal{X}})\mathbf{W}_b\mathbf{W}_b^{\mathrm{T}}\check{\mathbf{q}}_{\text{-}k}^{(m)}$ for each $b\in[B]$.

Essentially, we Bootstrap the mode-k fibres by choosing them randomly with replacement, and augment the vector of projection $\check{\mathbf{q}}_{\text{-}k}^{(m)}$ accordingly by pre-multiplying it with $\mathbf{W}_b^{\mathrm{T}}$. We control each row of $\mathbf{W}_b$ to contains at most 8 $\xi_i^{(b)}$'s, meaning that a fibre is at most chosen 8 times in each Bootstrap sample. This facilitates our theoretical proof of Theorem 5.3, although for all our simulations, a fibre is never chosen more than 8 times.

From here on, we drop the subscript $m+1$ for the ease of presentation. With the new projected data, we then create new covariance and correlation matrices:

$$\widetilde{\mathbf{\Sigma}}_{y,b}^{(k)} := T^{-1}\sum_{t=1}^{T}\mathbf{y}_{t,b}^{(k)}\mathbf{y}_{t,b}^{(k)\intercal}, \quad \widetilde{\mathbf{R}}_{y,b}^{(k)} := \text{diag}^{-1/2}(\widetilde{\mathbf{\Sigma}}_{y,b}^{(k)})\widetilde{\mathbf{\Sigma}}_{y,b}^{(k)}\text{diag}^{-1/2}(\widetilde{\mathbf{\Sigma}}_{y,b}^{(k)}), \;\; k\in[K], b\in[B].$$

THEOREM 5.3. *Let all the assumptions in Theorem 5.2 hold. Suppose for each $k\in[K]$, the elements in the unit vector $\mathbf{U}_{\text{-}k,(1)} =: (u_j)_{j\in[d_{\text{-}k}]}$ have the same moment structure up to the 4th order, and $E(u_i^2u_j^2) = d_{\text{-}k}^{-2}(1+o(1))$ for $i\neq j$ as $d_{\text{-}k}\to\infty$. Then Theorem 5.1 holds for $\mathbf{R}_{y,m+1}^{(k)}$ defined there but with $\check{\mathbf{q}}_{\text{-}k}^{(m)}$ in $\mathbf{\Sigma}_{y,m+1}^{(k)}$ replaced by $\mathbf{W}_b\mathbf{W}_b^{\intercal}\check{\mathbf{q}}_{\text{-}k}^{(m)}$. Theorem 5.2 holds also for $\widetilde{\mathbf{R}}_{y,b}^{(k)}$.*

The above theorem means that any procedures for finding the number of factors exploiting Theorem 5.1 and 5.2, should work for our Bootstrapped correlation matrix $\widetilde{\mathbf{R}}_{y,b}^{(k)}$ too. The assumption on $E(u_i^2u_j^2)$ is mild, since it is easily see that $E(u_i^2) = d_{\text{-}k}^{-1}$, so that at exact independence we have $E(u_i^2u_j^2) = d_{\text{-}k}^{-2}$. We are essentially assuming that the covariance among the u_i's are $o(d_{\text{-}k}^{-2})$, so that u_i and u_j are nearly uncorrelated.

For a constant C, we use $\eta_T = CT^{-1/2}$ for $K=2$ and $\eta_T = CT^{-1}$ for $K\geq 3$, and calculate

$$\widehat{r}_k^{(b)}(C) := \max\{j : \lambda_j(\widetilde{\mathbf{R}}_{y,b}^{(k)}) > 1+\eta_T,\; j\in[d_k]\}.$$

We propose to choose C with

$$\widehat{C} := \min_{C>0}\widehat{\text{var}}(\{\widehat{r}_k^{(b)}(C)\}_{b\in[B]}),$$

where $\widehat{\text{var}}(\{x_t\}_{t\in\mathcal{T}})$ is the sample variance of $\{x_t\}_{t\in\mathcal{T}}$. Finally, our estimator for r_k is defined to be

$$\check{r}_k := \text{Mode of } \{\widehat{r}_k^{(b)}(\widehat{C})\}_{b\in[B]}. \tag{5.4}$$

The intuition of $\widehat{C}$ and $\check{r}_k$ is as follows. If there are r_k factors for $\mathbf{A}_k$, then the first r_k eigenvalues of $\widetilde{\mathbf{R}}_{y,b}^{(k)}$ for each $b\in[B]$ should be approximately well-separated. Setting a large C will create a large threshold $1+\eta_T$ that is almost always lying in between $\lambda_j(\widetilde{\mathbf{R}}_{y,b}^{(k)})$ and $\lambda_{j+1}(\widetilde{\mathbf{R}}_{y,b}^{(k)})$ for some fixed $j\in[r_k]$ for each $b\in[B]$, so that $\widehat{\text{var}}(\{\widehat{r}_k^{(b)}(C)\}_{b\in[B]})$ will be small, or even equals 0.

However, if C is small such that $1+\eta_T$ is now in between $\lambda_j(\widetilde{\mathbf{R}}_{y,b}^{(k)})$ and $\lambda_{j+1}(\widetilde{\mathbf{R}}_{y,b}^{(k)})$ for some $j\in[d_k]/[r_k]$ and some $b\in[B]$, then we expect that this particular threshold will lie in between $\lambda_{j'}(\widetilde{\mathbf{R}}_{y,b}^{(k)})$ and $\lambda_{j'+1}(\widetilde{\mathbf{R}}_{y,b}^{(k)})$ for some $j'\neq j$ and some others $b\in[B]$, since all these eigenvalues are less than or equal to 1 by Theorem 5.1, and their variability is originated from the noise series only, making them less stable compared to when $j\in[r_k]$. Hence for a small enough C, we expect $\widehat{\text{var}}(\{\widehat{r}_k^{(b)}(C)\}_{b\in[B]})$ to be large. The range of values of C such that $1+\eta_T$ lies in between $\lambda_{r_k}(\widetilde{\mathbf{R}}_{y,b}^{(k)})$ and $\lambda_{r_k+1}(\widetilde{\mathbf{R}}_{y,b}^{(k)})$ for the majority of $b\in[B]$ will then include $\widehat{C}$. The definition of $\check{r}_k$ in (5.4) allows for variability arises from the noises and the r_k-th factor which can be weak and hence may not be detected in all Bootstrap samples.

Finally, in all the simulation settings in Section 6, we use $B=50$ Bootstrap samples. This is a safe number, since reducing it to 10 in fact hardly change the results in our simulation experiments.

6. Simulation Experiments. In this section, we conduct simulation experiments to compare the performances of our iterative projection estimators (PROJ) to other state-of-the-art competitors. The pre-averaging estimator (PRE) is presented and compared to PROJ in the supplement. We also test the performance of our proposed rank estimators (BCorTH) with bootstrapping of tensor fibres for tuning parameter selection. A set of NYC taxi traffic data is also analyzed in Section 6.4. An extra set of real data analysis is also presented in the supplement.

6.1. *Simulation settings.* For generating our data, we use model (3.1), with elements in μ being i.i.d. standard normal in each repetition of experiment. For $k \in [K]$, each factor loading matrix $\mathbf{A}_k$ is generated independently with $\mathbf{A}_k = \mathbf{B}_k \mathbf{R}_k$, where the elements in $\mathbf{B}_k \in \mathbb{R}^{d_k \times r_k}$ are i.i.d. $U(u_1, u_2)$, and $\mathbf{R}_k \in \mathbb{R}^{r_k \times r_k}$ is diagonal with the jth diagonal element being $d_k^{-\zeta_{k,j}}$, $0 \leq \zeta_{k,j} \leq 0.5$. Pervasive (strong) factors have $\zeta_{k,j} = 0$, while weak factors have $0 < \zeta_{k,j} \leq 0.5$.

The elements in $\mathcal{F}_t$ are independent standardized AR(5) with AR coefficients 0.7,0.3,-0.4,0.2 and -0.1. Same for the elements in $\mathcal{F}_{e,t}$ and $\boldsymbol{\epsilon}_t$ in (3.4), but their AR coefficients are (-0.7,-0.3,-0.4,0.2,0.1) and (0.8,0.4, -0.4,0.2,-0.1) respectively. The standard deviation of each element of $\boldsymbol{\epsilon}_t$ is randomly generated with i.i.d. $|\mathcal{N}(0,1)|$. Each entry of the matrices $\mathbf{A}_{e,k} \in \mathbb{R}^{d_k \times r_{e,k}}, k \in [K]$ is generated with i.i.d. standard normal, but has an independent probability of 0.7 being set exactly to 0. Each experiment is repeated 500 times. We consider the simulation settings (I), (II) and (III), with sub-settings (a) and (b), detailed below:

(Ia) All strong factors with $\zeta_{k,j} = 0$ for all k, j, and $u_1 = -2$, $u_2 = 2$ (rows of $\mathbf{A}_k$ sum to "small" magnitude (small s_k)).
(IIa) One strong factor with $\zeta_{k,1} = 0$ and $\zeta_{k,2} = 0.2$ for all k; $u_1 = -2$, $u_2 = 2$.
(IIIa) Two weak factors with $\zeta_{k,1} = 0.1$ and $\zeta_{k,2} = 0.2$ for all k; $u_1 = -2$, $u_2 = 2$.

Setting (Ib) to (IIIb) are the same as (Ia) to (IIIa) respectively, except that $u_1 = 0$, $u_2 = 2$, so that the row sums of $\mathbf{A}_k$ have large magnitude, leading to large s_k.

To test the performance of different estimation methods under heavy-tailed distributions, we consider two distributions for the innovation processes of $\mathcal{F}_t$, $\mathcal{F}_{e,t}$ and $\boldsymbol{\epsilon}_t$: 1) i.i.d. standard normal; 2) i.i.d. t_3. Thus, there are totally twelve profiles considered. For all profiles above, we set $r_k = r_{e,k} = 2$ for all k.

6.2. *Factor loading estimations.* We compare our iterative projection estimator (PROJ) in estimating the factor loading spaces with some state-of-art methods proposed by recent literature. All the twelve profiles in Section 6.1 and five settings of different dimensions are considered:

i. $K = 2, T = 100, d_1 = d_2 = 40$; ii. $K = 2, T = 200, d_1 = d_2 = 80$;
iii. $K = 3. T = 200, d_1 = d_2 = d_3 = 15$; iv. $K = 3. T = 200, d_1 = d_2 = d_3 = 25$;
v. $K = 4. T = 200, d_1 = d_2 = d_3 = d_4 = 15$.

When $K = 2$, the following methods designed for matrix-valued factor models are compared: The method of Wang, Liu and Chen (2019) is TOPOP in Chen, Yang and Zhang (2022), but we omit its results since it performs much worse than iTIPUP in Han et al. (2020), which is the best one among the same type of estimators. The α-PCA estimator of Chen and Fan (2021) is implemented with $\alpha = 0$ (the performances for $\alpha \in \{-1, 0, 1\}$ are comparable according to Yu et al. (2022)). The projection method of Yu et al. (2022) and He, Li and Trapani (2022) are referred to as PE (which is in the same spirit as HOOI). In addition, we also consider some robust procedures, including the robust tensor factor analysis (RTFA) proposed by He et al. (2022) and He et al. (2023a), and the Matrix Kendall's tau

(MRTS) by He, Li and Trapani (2022). For all the above methods which involve iterations, we set the number of iterations to be 30.

For settings with $K = 3$, we do not include α-PCA and MRTS, since they are only designed for $K = 2$. For the setting with $K = 4$, we further exclude RTFA in comparison, since it requires too much computational time, as can be observed in Table 1 in the supplement.

Figure 2 and 3 show the logarithm of estimation errors of $\mathbf{A}_1$ under the first four different settings of dimensions, with normally and t_3 distributed errors, respectively. The results with $K = 4$ is included in Section 1.2 in our supplement. It can be seen that our iterative projection estimator (PROJ) generally outperforms all competitors in all settings and dimensions we consider, and is at least on par with other competitors.

More specifically, when $K = 2$, all methods perform reasonably well in sub-setting (a), but they all perform poorly in sub-setting (b) except our iterative projection estimator (PROJ). The only difference between sub-setting (a) and (b) is that the mean of each element of $\mathbf{A}_k$ in sub-setting (a) is 0 while it is non-zero in sub-setting (b). Whenever the mean of the elements are not 0, the pre-averaging estimator, and hence the iterative projection estimator, can take advantage since pre-averaging is based on summing rows of $\mathbf{A}_1$ (for estimating $\mathbf{A}_2$) or $\mathbf{A}_2$ (for estimating $\mathbf{A}_1$).

Regarding factor strengths, the advantage of PROJ is more notable in Setting (II) and (III), when other methods tend to give poorer estimates in the presence of weak factors in these settings. Our method is also robust to heavy-tailed errors, and perform better than the robust procedure RTFA and MRTS in all scenarios. When $K = 3$ (and 4 as well; see Section 1.2 in our supplement), most methods take advantage of a larger K and perform better. Our iterative projection estimator still performs better than, or at least on par with all competitors. For computation time, see Section 1.2 of our supplement.

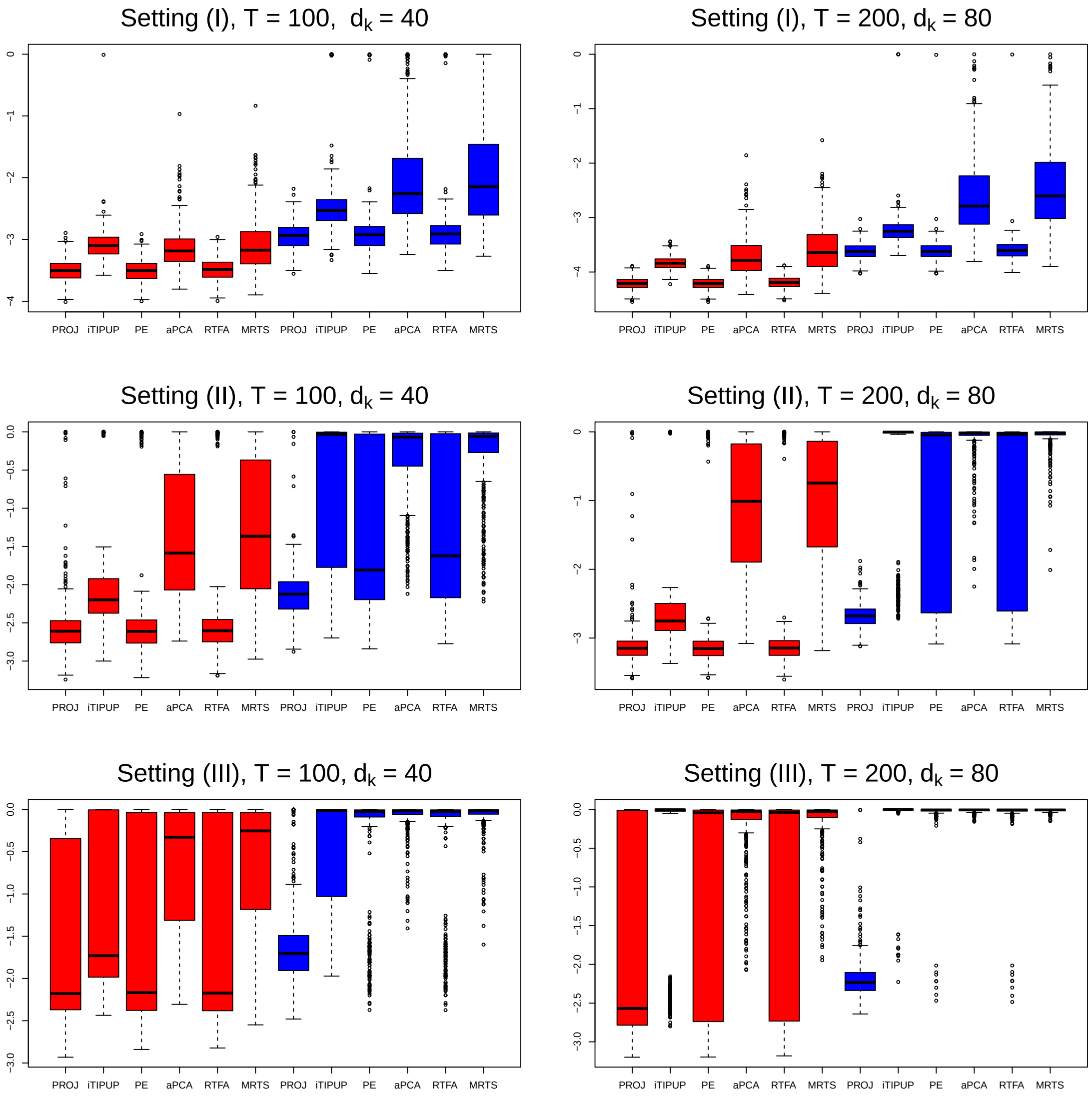
6.3. *Core tensor rank estimations.* We compare the performance of our BCorTh with other competitors for estimating the rank of core tensors. The methods we consider include iTIP-ER by Han, Zhang and Chen (2022a), α-PCA-ER by Chen and Fan (2021), PE-ER by Yu et al. (2022) and He, Li and Trapani (2022), RTFA-ER by He et al. (2022) and He et al. (2023a), and MRTS-ER by He, Li and Trapani (2022). Most of these methods are based on the spirit of eigenvalue-ratio criteria of the (adjusted) sample covariance matrices, which are defined differently in their corresponding processes of factor loading estimations. For ease of presentation, the following three set of dimensions are considered:

i. $K = 2,\ T = 100,\ d_1 = d_2 = 40$; ii. $K = 2,\ T = 200,\ d_1 = d_2 = 80$; iii. $K = 3.\ T = 200,\ d_k = 25$.

Table 1 records the correct proportion over 500 repetitions of different rank estimators under different settings and dimensions. For BCorTh, we set the number of bootstrapped samples to be $B = 50$ for settings with $K = 2$, and $B = 10$ for settings with $K = 3$. We have tested that reducing B from 50 to 10 does not significantly change the results of BCorTh. Also, for $K = 3$, we do not report the results for α-PCA-ER and MRTS-ER since they are only designed for matrix time series ($K = 2$).

From Table 1, all rank estimators perform better when T, d_k or K increases, and BCorTh outperforms all competitors in every setting and dimension we consider. When $K = 2$, it is obvious that all methods, except BCorTh, perform quite poorly in Setting (II) and (III) when weak factors are present (especially in sub-setting (b)), while BCorTh can still give relatively good performances. MRTS-ER and α-PCA-ER give extremely poor estimates in all settings except (Ia). BCorTh is robust as well, since changing the error distribution from normal to t_3 does not have large effects in its estimation accuracy.

When $K = 3$, the accuracy of all estimators increases, and BCorTh still gives the best performances among all competitors.


Setting (I), T = 100, d_k = 40

Setting (I), T = 200, d_k = 80

Setting (II), T = 100, d_k = 40

Setting (II), T = 200, d_k = 80

Setting (III), T = 100, d_k = 40

Setting (III), T = 200, d_k = 80

Fig 2: Plot of estimation error $\left\|\widehat{\mathbf{Q}}_1\widehat{\mathbf{Q}}_1^\top - \mathbf{U}_1\mathbf{U}_1^\top\right\|$ (in log-scale) for $K = 2$, normally distributed errors. In each panel, the left six boxplots (in red) represent sub-setting (a), while the right six boxplots (in blue) represent sub-setting (b).

6.4. *NYC taxi traffic.* We analyze taxi traffic pattern in New York city. The data includes all individual taxi rides operated by Yellow Taxi within New York City, published at

https://www1.nyc.gov/site/tlc/about/tlc-trip-record-data.page.

To simplify the discussion, we only consider rides within Manhattan Island. The dataset contains 1.1 billion trip records within the period of January 1, 2011 to December 31, 2021. Each trip record includes fields capturing pick-up and drop-off dates/times, pick-up and drop-off locations, trip distances, itemized fares, rate types, payment types, and driver-reported passenger counts. Our study focuses on the pick-up and drop-off dates/times, and pick-up and drop-off locations of each ride.

The pick-up and drop-off locations in Manhattan are coded according to 69 predefined zones and we will use them to classify the pick-up and drop-off locations. Furthermore, we divide each day into 24 hourly periods, with the first hourly period from 0am to 1am.

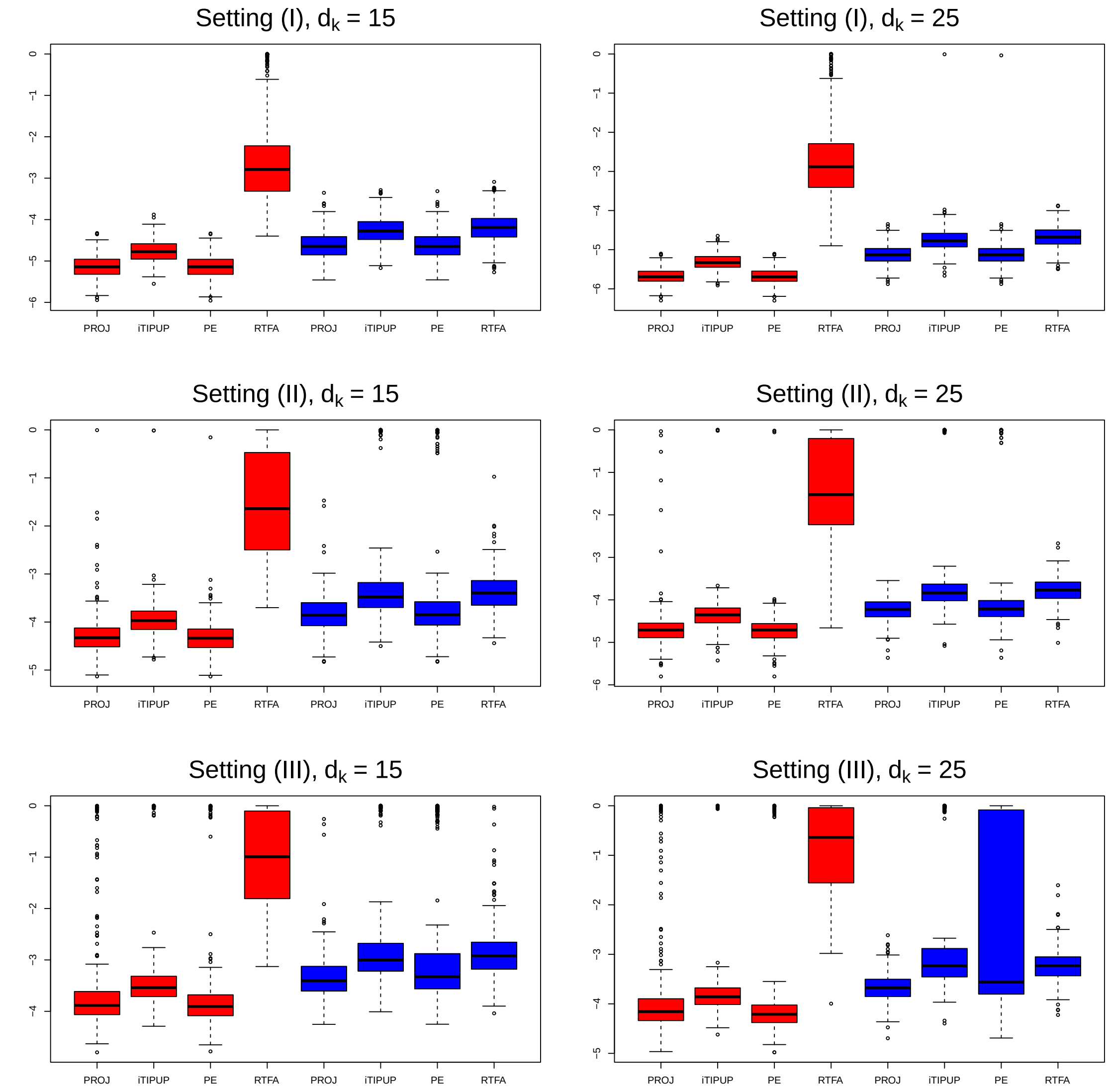


Fig 3: Plot of estimation error $\left\|\widehat{\mathbf{Q}}_1\widehat{\mathbf{Q}}_1^\top - \mathbf{U}_1\mathbf{U}_1^\top\right\|$ (in log-scale) for $K = 3, T = 200$, t_3-distributed errors. In each panel, the left four boxplots (in red) represent sub-setting (a), while the right four boxplots (in blue) represent sub-setting (b).

The total number of rides moving among the zones within each hour is recorded, yielding a $\mathcal{X}_t \in \mathbb{R}^{69\times 69\times 24}$ tensor for each day. More specifically, $x_{i1,i2,i3,t}$ is the number of trips from zone i_1 (the pick-up zone) to zone i_2 (the drop-off zone) and the pickup time is within the i_3-th hourly period on day t. We consider business day and non-business day separately. Hence we will analyze two tensor time series. The business-day series is 2770 days long, and the non-business-day series is 1248 days long, within the period of January 1, 2011 to December 31, 2021.

We first estimate the rank of the core tensors using BCorTh as well as other state-of-the-art methods. BCorTh gives $(\widehat{r}_1, \widehat{r}_2, \widehat{r}_3) = (3, 3, 2)$ for business-day series, and $(\widehat{r}_1, \widehat{r}_2, \widehat{r}_3) = (3, 2, 2)$ for non-business-day series, while $(\widehat{r}_1, \widehat{r}_2, \widehat{r}_3) = (1, 1, 1)$ for iTIP-ER, PE-ER and RTFA-ER. However, based on our common knowledge and previous analysis conducted by Chen, Yang and Zhang (2022), $(\widehat{r}_1, \widehat{r}_2, \widehat{r}_3) = (1, 1, 1)$ is obviously not a reasonable choice for

Setting	BCorTh		iTIP-ER		PE-ER		α-PCA-ER		RTFA-ER		MRTS-ER	
	$\mathcal{N}$	t_3	$\mathcal{N}$	t_3	$\mathcal{N}$	t_3	$\mathcal{N}$	t_3	$\mathcal{N}$	t_3	$\mathcal{N}$	t_3
$K=2,\ T=100,\ d_1=d_2=40$												
(Ia)	.994	.988	.894	.866	.892	.878	.842	.810	.894	.884	.842	.810
(Ib)	.998	1.000	.830	.794	.908	.896	.014	.012	.906	.898	.040	.030
(IIa)	.994	.966	.754	.640	.762	.690	.010	.038	.768	.702	.026	.022
(IIb)	.954	.928	.070	.084	.126	.080	.014	.046	.092	.070	.026	.056
(IIIa)	.758	.684	.556	.482	.334	.408	.054	.092	.320	.372	.048	.068
(IIIb)	.772	.712	.108	.202	.052	.116	.122	.180	.048	.106	.128	.182
$K=2,\ T=200,\ d_1=d_2=80$												
(Ia)	.994	.990	.926	.910	.768	.854	.944	.904	.768	.842	.768	.804
(Ib)	.998	1.000	.966	.956	.073	.898	.006	.012	.686	.884	.020	.022
(IIa)	.992	.972	.814	.796	.438	.632	.000	.010	.406	.602	.000	.000
(IIb)	.998	.998	.258	.176	.332	.230	.004	.024	.296	.230	.004	.016
(IIIa)	.594	.620	.188	.292	.014	.092	.000	.010	.016	.092	.000	.010
(IIIb)	.978	.968	.096	.074	.078	.082	.060	.074	.070	.080	.028	.054
$K=3,\ T=200,\ d_1=d_2=25$												
(Ia)	1.000	1.000	.996	.990	.992	.980	/	/	.776	.732	/	/
(Ib)	1.000	.998	.998	.986	.972	.982	/	/	.998	1.000	/	/
(IIa)	1.000	.988	.988	.968	.834	.854	/	/	.106	.072	/	/
(IIb)	.994	.992	.988	.968	.948	.940	/	/	.948	.920	/	/
(IIIa)	.930	.856	.910	.866	.522	.544	/	/	.100	.056	/	/
(IIIb)	.996	.996	.920	.868	.764	.738	/	/	.804	.760	/	/

TABLE 1

Correct Proportion ($(\widehat{r}_1,\widehat{r}_2)=(2,2)$ for $K=2$, $(\widehat{r}_1,\widehat{r}_2,\widehat{r}_3)=(2,2,2)$ for $K=3$) of rank estimation under different settings, dimensions and error distributions ($\mathcal{N}$ for normally distributed errors, t_3 for t_3 distributed errors).

the rank of the core tensor, since a single factor can hardly be sufficient to reveal all traffic patterns. It is very likely that all of iTIP-ER, PE-ER and RTFA-ER fail to detect the weak factors in both time series, since these methods are designed to analyze pervasive factors only. For ease of presentation and comparison, we use $(\widehat{r}_1,\widehat{r}_2,\widehat{r}_3)=(3,3,2)$ for both business-day and non-business-day series to estimate their factor loadings, and present the results of our iterative projection estimator.

Figure 4 and 5 show the heatmaps of the loading matrices $\mathbf{A}_1$ (pick-up locations) for the business day and non-business day series, respectively. It is seen that during business days, the midtown/Times square area (tourism and office buildings) is heavily loaded on Factor 1, east village/lower east (arts, music venues and restaurants) on Factor 2 and upper east side (affluent neighborhoods and museums) on Factor 3. For non-business days, the overall pattern for the three factors is generally similar, but with some non-negligible differences. The area around Penn Station (large transportation hub) loads extremely heavily in Factor 2, while its loading is much lighter than the midtown center and midtown east for business day series, where a lot of office buildings locate.

Figure 6 and 7 show the loading matrices $\mathbf{A}_2$ (drop-off locations) for the business day and non-business day series, respectively. For both business days and non-business days, the drop-off factor matrices are quite similar to their pick-up factors. Similarly, the area around Penn Station is heavily loaded in non-business days, but is overshadowed by midtown center in business days. In addition, in Factor 1 of non-business days series, west village (arts, music venues and theatres) loads heavily together with east village.

Table 2 and 3 show the loading matrices $\mathbf{A}_3$ (time of day) for business days and non-business days, respectively. For ease of presentation, we show the estimated loading matrices after a varimax rotation, scaled by 30 for a cleaner view. For business days, it can be seen that day-time business hour (9am to 4pm) and evening hours (7pm to 12am) load heavily on Factor 1, while morning rush-hours (6am to 9am), evening rush-hours (5pm to 7pm) and night life hours (0am to 2am) load heavily on Factor 2. For non-business days, the patterns of estimated factors are significantly different: Evening hours from 6pm to 1am load heavily on Factor 1, while late-night hours from 1am to 5am load heavily on Factor 2. The different factor loadings reveal the difference between people's travelling habits in business days and non-business days. During non-business days, morning (and evening) rush-hours and day-time business hours no longer appear in the factors, while people tend to travel more frequently by taxi at evening and at night, and their night life lasts to much later hours than in the business days.

	0am		2		4		6		8		10		12pm		2		4		6		8		10	12am
1	6	3	2	1	1	0	1	4	7	9	8	7	7	7	7	7	6	4	6	8	9	8	8	7
2	12	7	4	2	1	-2	-11	-16	-8	0	3	-1	-2	0	-2	0	3	-7	-9	-2	3	0	3	8

TABLE 2
Estimated loading matrix $\mathbf{A}_3$ for hour of day fibre, business day, after rotation and scaling. Magnitudes larger than 7 are highlighted in red.

	0am		2		4		6		8		10		12pm		2		4		6		8		10	12am
1	9	-1	-1	-1	-2	-1	1	2	3	5	5	5	5	5	5	5	4	6	8	9	9	9	10	12
2	0	17	14	13	10	5	1	-1	-1	-2	0	2	4	4	5	4	4	2	0	0	2	0	-2	-4

TABLE 3
Estimated loading matrix $\mathbf{A}_3$ for hour of day fibre, non-business day, after rotation and scaling. Magnitudes larger than 7 are highlighted in red.

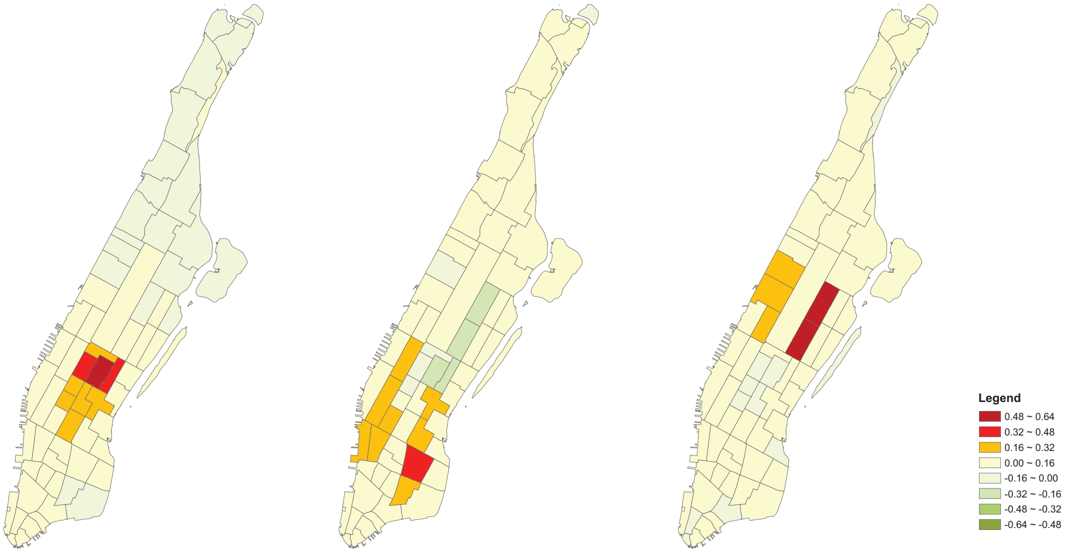


Fig 4: Loadings on three pickup factors for business day series.

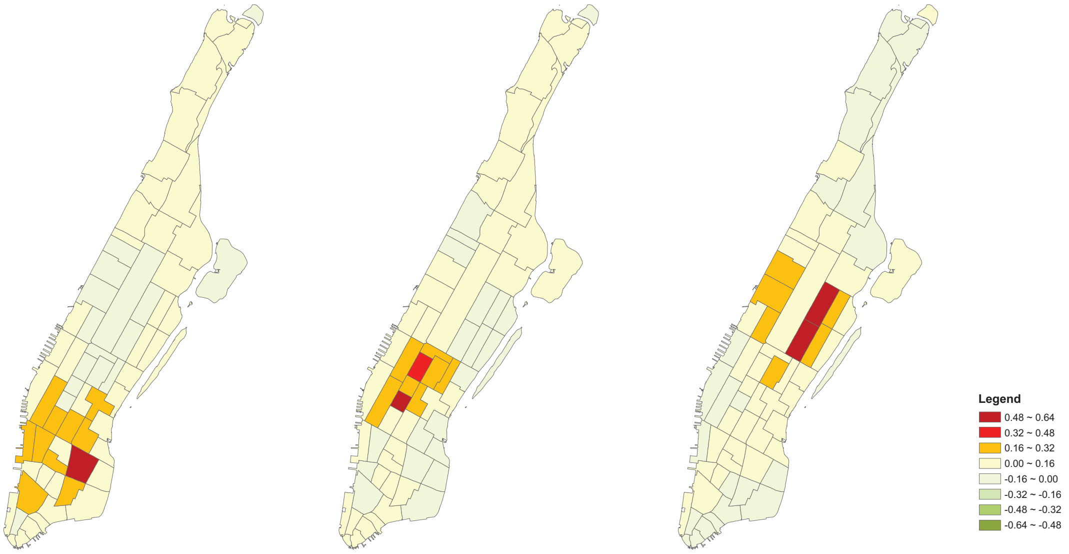


Fig 5: Loadings on three pickup factors for non-business day series.

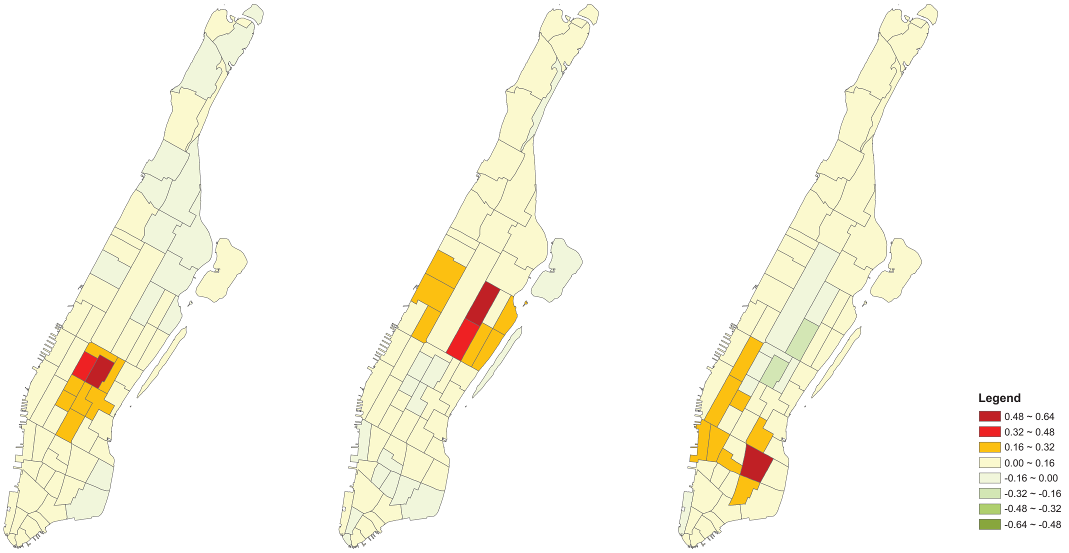


Fig 6: Loadings on three dropoff factors for business day series.

SUPPLEMENTARY MATERIAL

Proof of Theorems, Extra Simulations and Data Analysis
Extra simulation results, analysis of a set of matrix-valued portfolio return data, a brief introduction to the use of the R package `TensorPreAve` and the proofs of all the theorems in this paper can be found at http://stats.lse.ac.uk/lam/Supp-PROJ.pdf. The R codes to replicate the results of our simulation experiments and data analyses can be found here.

REFERENCES

BAI, J. and NG, S. (2002). Determining the Number of Factors in Approximate Factor Models. *Econometrica* **70** 191–221.

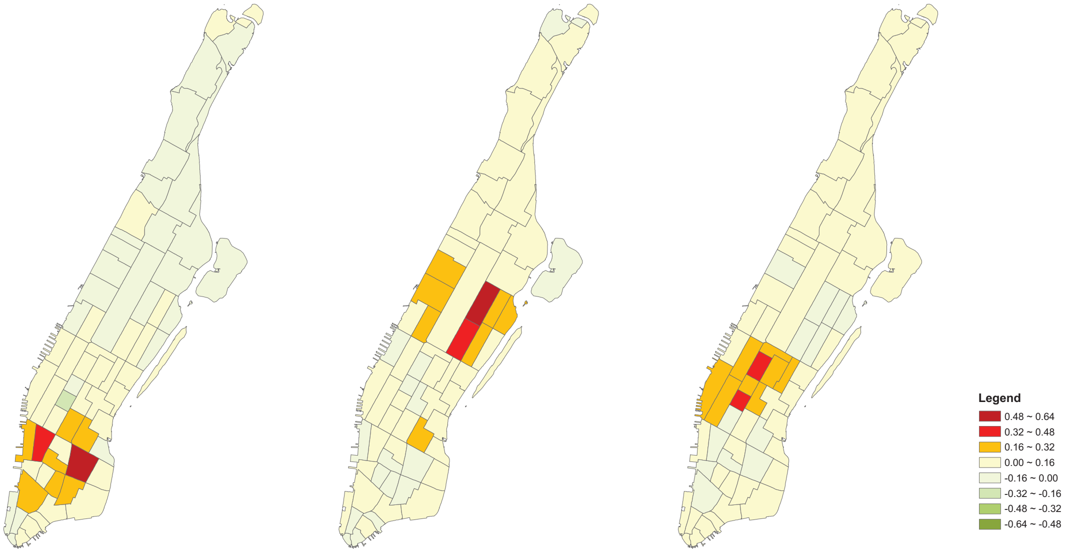


Fig 7: Loadings on three dropoff factors for non-business day series.

BAI, J. and NG, S. (2021). Approximate Factor Models with Weaker Loadings Papers No. 2109.03773, arXiv.org.

BARIGOZZI, M., HE, Y., LI, L. and TRAPANI, L. (2023). Robust estimation of large factor models for tensor-valued time series. *arXiv:2303.18163*.

CHEN, E. Y. and FAN, J. (2021). Statistical inference for high-dimensional matrix-variate factor models. *Journal of the American Statistical Association* **118** 1038-1055. https://doi.org/10.1080/01621459.2021.1970569

CHEN, R., YANG, D. and ZHANG, C.-H. (2022). Factor Models for High-Dimensional Tensor Time Series. *Journal of the American Statistical Association* **117** 94-116.

CHEN, E. Y., XIA, D., CAI, C. and FAN, J. (2020a). Semiparametric Tensor Factor Analysis by Iteratively Projected SVD. *arXiv preprint arXiv:2007.02404*.

CHEN, E. Y., YUN, X., CHEN, R. and YAO, Q. (2020b). Modeling Multivariate Spatial-Temporal Data with Latent Low-Dimensional Dynamics. *arXiv preprint arXiv:2002.01305*.

CHEN, R., HAN, Y., LI, Z., XIAO, H., YANG, D. and YU, R. (2022). Analysis of Tensor Time Series: tensorTS. Submitted.

CHENG, C., WEI, Y. and CHEN, Y. (2021). Tackling Small Eigen-Gaps: Fine-Grained Eigenvector Estimation and Inference under Heteroscedastic Noise. *IEEE Transactions on Information Theory* **67** 7380–7419. Publisher Copyright: © 1963-2012 IEEE. https://doi.org/10.1109/TIT.2021.3111828

FAN, J., GUO, J. and ZHENG, S. (2022). Estimating Number of Factors by Adjusted Eigenvalues Thresholding. *Journal of the American Statistical Association* **117** 852-861. https://doi.org/10.1080/01621459.2020.1825448

FREYALDENHOVEN, S. (2022). Factor models with local factors âĂŤ Determining the number of relevant factors. *Journal of Econometrics* **229** 80-102. https://doi.org/10.1016/j.jeconom.2021.04.006

HAN, Y., ZHANG, C. H. and CHEN, R. (2022a). Rank Determination in Tensor Factor Model. *Electronic Journal of Statistic* **16** 1726–1803.

HAN, Y., ZHANG, C.-H. and CHEN, R. (2022b). CP Factor Model for Dynamic Tensors. Under revision at the Journal of the Royal Statistical Society, Series B (Statistical Methodology).

HAN, Y. and ZHANG, C.-H. (2023). Tensor Principal Component Analysis in High Dimensional CP Models. *IEEE Transactions on Information Theory* **69** 1147-1167. https://doi.org/10.1109/TIT.2022.3203972

HAN, Y., CHEN, R., YANG, D. and ZHANG, C.-H. (2020). Tensor Factor Model Estimation by Iterative Projection. *arXiv: Methodology*.

HARTIGAN, J. A. (2014). Bounding the maximum of dependent random variables. *Electronic Journal of Statistics* **8** 3126 – 3140. https://doi.org/10.1214/14-EJS974

HE, Y., LI, L. and TRAPANI, L. (2022). Statistical Inference for Large-dimensional Tensor Factor Model by Weighted/Unweighted Projection. *arXiv:2206.09800*.

HE, Y., WANG, Y., YU, L., ZHOU, W. and ZHOU, W.-X. (2022). Matrix Kendall's tau in High-dimensions: A Robust Statistic for Matrix Factor Model. *arXiv:2207.09633*.

HE, Y., KONG, X., YU, L., ZHANG, X. and ZHAO, C. (2023a). Matrix factor analysis: From least squares to iterative projection. *Journal of Business and Economic Statistics* **0** 1 - 13. https://doi.org/10.1080/07350015.2023.2191676

HE, Y., KONG, X.-B., LIU, D. and ZHAO, R. (2023b). Robust Statistical Inference for Large-dimensional Matrix-valued Time Series via Iterative Huber Regression. *arXiv:2306.03317*.

KOLDA, T. G. and BADER, B. W. (2009). Tensor Decompositions and Applications. *SIAM Review* **51** 455-500. https://doi.org/10.1137/07070111X

LAM, C., YAO, Q. and BATHIA, N. (2011). Estimation of latent factors for high-dimensional time series. *Biometrika* **98** 901-918. https://doi.org/10.1093/biomet/asr048

LIU, X. and CHEN, E. Y. (2022). Identification and estimation of threshold matrix-variate factor models. *Scandinavian Journal of Statistics* **49** 1383-1417. https://doi.org/10.1111/sjos.12576

LIU, T., YUAN, M. and ZHAO, H. (2022). Characterizing Spatiotemporal Transcriptome of the Human Brain Via Low-Rank Tensor Decomposition. *Statistics in Biosciences*. https://doi.org/10.1007/s12561-021-09331-5

STOCK, J. H. and WATSON, M. W. (2002). Forecasting Using Principal Components from a Large Number of Predictors. *Journal of the American Statistical Association* **97** 1167–1179.

TAO, M., SU, J. and WANG, L. (2019). Land cover classification of PolSAR image using tensor representation and learning. *Journal of Applied Remote Sensing* **13** 016516. https://doi.org/10.1117/1.JRS.13.016516

WANG, D., LIU, X. and CHEN, R. (2019). Factor models for matrix-valued high-dimensional time series. *Journal of Econometrics* **208** 231-248. Special Issue on Financial Engineering and Risk Management. https://doi.org/10.1016/j.jeconom.2018.09.013

YOKOTA, T., LEE, N. and CICHOCKI, A. (2017). Robust Multilinear Tensor Rank Estimation Using Higher Order Singular Value Decomposition and Information Criteria. *IEEE Transactions on Signal Processing* **65** 1196-1206. https://doi.org/10.1109/TSP.2016.2620965

YU, L., HE, Y., KONG, X. and ZHANG, X. (2022). Projected estimation for large-dimensional matrix factor models. *Journal of Econometrics* **229** 201âĂŞ217. https://doi.org/10.1016/j.jeconom.2021.04.001

ZHANG, A. R. and XIA, D. (2018). Tensor SVD: Statistical and Computational Limits. *IEEE Transactions on Information Theory* **64** 7311-7338.