Tail-robust Factor Modelling of Vector and Tensor Time Series in High Dimensions

Presenter: Ben-Zheng Li

March 14, 2026

Reference: Matteo Barigozzi, Haeran Cho, Hyeyoung Maeng,Tail-robust factor modelling of vector and tensor time series in high dimensions, *arXiv*,2025.

Background: Vector Factor Model (VFM)

- **High-dimensional Data**: Let $\mathcal{X}_t \in \mathbb{R}^{n_1 \times \cdots \times n_d}$ be a d-order tensor observed at $t = 1, \ldots, T$.
- **Vectorization**: The conventional approach flattens the tensor $\mathcal{X}_t$ into a long vector $\mathbf{x}_t = \text{vec}(\mathcal{X}_t) \in \mathbb{R}^N$, where:

$$N = \prod_{j=1}^{d} n_j = n_1 \times n_2 \times \cdots \times n_d \tag{1}$$

- **Model Specification**: The data is assumed to follow a linear factor structure:

$$\mathbf{x}_t = \boldsymbol{\Lambda} \mathbf{f}_t + \mathbf{e}_t \tag{2}$$

where:
  - $\boldsymbol{\Lambda} \in \mathbb{R}^{N \times R}$ is the **loading matrix**.
  - $\mathbf{f}_t \in \mathbb{R}^R$ is the vector of **common factors**.
  - $\mathbf{e}_t \in \mathbb{R}^N$ is the idiosyncratic component.

Background: Drawbacks of Vectorization

Vectorizing a tensor leads to two major drawbacks:

- **The Curse of Dimensionality**:
  - The number of parameters in $\mathbf{\Lambda}$ is $N \times R = (\prod n_j) \times R$.
  - Problem: As the order d increases, N grows exponentially. Estimation becomes numerically unstable and computationally expensive.

Background: Drawbacks of Vectorization

Vectorizing a tensor leads to two major drawbacks:

- **The Curse of Dimensionality**:
  - The number of parameters in $\mathbf{\Lambda}$ is $N \times R = (\prod n_j) \times R$.
  - Problem: As the order d increases, N grows exponentially. Estimation becomes numerically unstable and computationally expensive.
- **Loss of Structural Information**:
  - Flattening treats each entry as an independent coordinate.

Background: Drawbacks of Vectorization

Vectorizing a tensor leads to two major drawbacks:

- **The Curse of Dimensionality**:
  - The number of parameters in $\mathbf{\Lambda}$ is $N \times R = (\prod n_j) \times R$.
  - Problem: As the order d increases, N grows exponentially. Estimation becomes numerically unstable and computationally expensive.
- **Loss of Structural Information**:
  - Flattening treats each entry as an independent coordinate.
- **The Tensor Solution**: By preserving $\mathcal{X}_t$ as a tensor, this paper uses a **Tucker-type** structure to reduce parameters to $O(\sum_{j=1}^{d} n_j r_j)$, effectively capturing mode-specific correlations.

Background: Tensor Factor Model (TFM)

This paper considers the **Tucker-type** tensor factor model :

$$\mathcal{X}_t = \mathcal{F}_t \times_1 \mathbf{A}_1 \times_2 \mathbf{A}_2 \cdots \times_d \mathbf{A}_d + \mathcal{E}_t \tag{3}$$

where the components are defined as:

- $\mathcal{X}_t \in \mathbb{R}^{n_1 \times \cdots \times n_d}$: Observed d-order tensor at time t.
- $\mathcal{F}_t \in \mathbb{R}^{r_1 \times \cdots \times r_d}$: **Core factor tensor** representing common shocks.
- $\mathbf{A}_j \in \mathbb{R}^{n_j \times r_j}$: **Mode-$j$ loading matrix**. We assume $\mathbf{A}_j' \mathbf{A}_j = \mathbf{I}_{r_j}$ for identifiability.
- $\mathcal{E}_t$: **Idiosyncratic component** (异质成分).
- $\times_j$: **Mode-j product**.

Background: Tensor Factor Model (TFM)

This paper considers the **Tucker-type** tensor factor model :

$$\mathcal{X}_t = \mathcal{F}_t \times_1 \mathbf{A}_1 \times_2 \mathbf{A}_2 \cdots \times_d \mathbf{A}_d + \mathcal{E}_t \tag{3}$$

where the components are defined as:

- $\mathcal{X}_t \in \mathbb{R}^{n_1 \times \cdots \times n_d}$: Observed d-order tensor at time t.
- $\mathcal{F}_t \in \mathbb{R}^{r_1 \times \cdots \times r_d}$: **Core factor tensor** representing common shocks.
- $\mathbf{A}_j \in \mathbb{R}^{n_j \times r_j}$: **Mode-$j$ loading matrix**. We assume $\mathbf{A}_j' \mathbf{A}_j = \mathbf{I}_{r_j}$ for identifiability.
- $\mathcal{E}_t$: **Idiosyncratic component** (异质成分).
- $\times_j$: **Mode-j product**.

Advantage

While VFM requires $(\prod n_j) \times R$ parameters, TFM only requires $\sum_{j=1}^{d} n_j r_j$ parameters in the loading matrices, where $r_j \ll n_j$.

Properties of the Idiosyncratic Component $\mathcal{E}_t$

The core challenge of this paper lies in the "unfriendly" nature of the noise $\mathcal{E}_t = \{e_{t,\mathbf{i}}\}$:

- **Heavy-Tailedness**:
  - Traditional TFM assumes finite 4th moments ($\mathbb{E}|e_{t,\mathbf{i}}|^4 < \infty$).
  - This paper relax this to $(2+2\epsilon)$-**th moments**:

$$\mathbb{E}|e_{t,\mathbf{i}}|^{2+2\epsilon} < \infty \quad \text{for some } \epsilon \in (0,1] \tag{4}$$

  - Impact: High probability of outliers that "contaminate" the covariance structure.

Properties of the Idiosyncratic Component $\mathcal{E}_t$

The core challenge of this paper lies in the "unfriendly" nature of the noise $\mathcal{E}_t = \{e_{t,\mathbf{i}}\}$:

- **Heavy-Tailedness**:
  - Traditional TFM assumes finite 4th moments ($\mathbb{E}|e_{t,\mathbf{i}}|^4 < \infty$).
  - This paper relax this to $(2+2\epsilon)$-**th moments**:

$$\mathbb{E}|e_{t,\mathbf{i}}|^{2+2\epsilon} < \infty \quad \text{for some } \epsilon \in (0,1] \tag{4}$$

  - Impact: High probability of outliers that "contaminate" the covariance structure.
- **Weak Dependence**:
  - **Temporal**: Allows for serial correlation over time t.
  - **Spatial**: Allows for cross-sectional correlation within each mode.

Properties of the Idiosyncratic Component $\mathcal{E}_t$

The core challenge of this paper lies in the "unfriendly" nature of the noise $\mathcal{E}_t = \{e_{t,\mathbf{i}}\}$:

- **Heavy-Tailedness**:
  - Traditional TFM assumes finite 4th moments ($\mathbb{E}|e_{t,\mathbf{i}}|^4 < \infty$).
  - This paper relax this to $(2+2\epsilon)$-**th moments**:

$$\mathbb{E}|e_{t,\mathbf{i}}|^{2+2\epsilon} < \infty \quad \text{for some } \epsilon \in (0,1] \tag{4}$$

  - Impact: High probability of outliers that "contaminate" the covariance structure.
- **Weak Dependence**:
  - **Temporal**: Allows for serial correlation over time t.
  - **Spatial**: Allows for cross-sectional correlation within each mode.
- **The Failure of Standard PCA**: When ϵ is small, the sample second-moment matrices are no longer stable, making the standard loading space estimators $\widehat{\mathbf{A}}_j$ inconsistent or highly biased.

Methodology: Data Truncation

To achieve robustness against heavy tails, this paper first preprocess the observed tensor $\mathcal{X}_t$ using an element-wise **truncation operator**:

Definition (Truncation Operator)

For a given threshold $\tau > 0$, define the function $\psi_\tau(x)$ as:

$$\psi_\tau(x) = \mathsf{sgn}(x)\min(|x|, \tau) \tag{5}$$

The truncated observations are denoted as $\widehat{\mathcal{X}}_t = \{\widehat{x}_{t,\mathbf{i}}\}$, where:

$$\widehat{x}_{t,\mathbf{i}} = \psi_{\tau_{n,T}}(x_{t,\mathbf{i}}) \tag{6}$$

- **Intuition**: It "pulls back" extreme outliers to the boundaries $[-\tau, \tau]$, effectively forcing the data to have finite moments of all orders.
- **Tuning Parameter**: The threshold $\tau_{n,T}$ is a crucial parameter that diverges to ∞ as $n, T \to \infty$.
- Advantage: This operation allows us to handle heavy tails without changing the loss function to a non-linear one like

Methodology: Algorithm 1 (Part I)

Algorithm 1: TIP — Truncation & Initialization

Input: Observed tensor $\mathcal{X}_t$, ranks $\{r_k\}$, truncation level τ, iterations $N = 2$.
Output: Estimated loading matrices $\{\widehat{\mathbf{A}}_k\}_{k=1}^{d}$.

1. **Data Truncation**
   Preprocess observations to obtain the truncated tensor $\widehat{\mathcal{X}}_t = \{\widehat{x}_{t,\mathbf{i}}\}$:
   $$\widehat{x}_{t,\mathbf{i}} = \mathrm{sgn}(x_{t,\mathbf{i}}) \min(|x_{t,\mathbf{i}}|, \tau)$$

2. **Initialization**
   For each mode k, compute initial loadings $\widehat{\mathbf{A}}_k^{(0)}$ as the top r_k eigenvectors of:
   $$\widehat{\mathbf{\Sigma}}_k = \frac{1}{T} \sum_{t=1}^{T} \mathrm{Mat}_k(\widehat{\mathcal{X}}_t) \mathrm{Mat}_k(\widehat{\mathcal{X}}_t)^\top$$

Methodology: Algorithm 1 (Part II)

Algorithm 1: TIP — Iterative Refinement

3. Iterative Refinement

For $\iota = 1, \ldots, N$:

a) **Projection:** Denoise by projecting the **original** $\mathcal{X}_t$ onto the estimated loading spaces of *other* modes:

$$\mathcal{Y}_{t,k}^{(\iota)} = \mathcal{X}_t \times_1 \widehat{\mathbf{A}}_1^{(\iota-1)\top} \cdots \times_{k-1} \widehat{\mathbf{A}}_{k-1}^{(\iota-1)\top} \times_{k+1} \widehat{\mathbf{A}}_{k+1}^{(\iota-1)\top} \cdots \times_d \widehat{\mathbf{A}}_d^{(\iota-1)\top}$$

b) **Update:** Set $\widehat{\mathbf{A}}_k^{(\iota)}$ as the top r_k eigenvectors of the sample covariance matrix of $\mathrm{Mat}_k(\mathcal{Y}_{t,k}^{(\iota)})$.

4. Output

Return final estimated loading matrices $\widehat{\mathbf{A}}_k = \widehat{\mathbf{A}}_k^{(N)}$.

Theoretical Note: $N = 2$ iterations suffice for asymptotic normality.

Theoretical Properties: Convergence Rates

Let $d(\widehat{\mathbf{A}}_k, \mathbf{A}_k) = \|\widehat{\mathbf{A}}_k \widehat{\mathbf{A}}_k^\top - \mathbf{A}_k \mathbf{A}_k^\top\|_F$ measure the distance between the estimated and true loading spaces.

Theorem (Convergence of TIP Estimator)

Under the $(2+2\epsilon)$-th moment assumption and suitable truncation $\tau \asymp (nT)^{1/(2+2\epsilon)}$, the TIP estimator $\widehat{\mathbf{A}}_k$ satisfies:

$$d(\widehat{\mathbf{A}}_k, \mathbf{A}_k) = O_p\left(\frac{1}{\sqrt{T}} + \frac{1}{\sqrt{n^{(k)}}}\right) \cdot \textit{Log Factors} \tag{7}$$

where $n^{(k)} = \prod_{j \neq k} n_j$ is the dimension of the rest of the tensor.

- **Optimal Rate**: The estimator achieves the same convergence rate as the non-heavy-tailed case (up to logarithmic terms).
- **The Role of ϵ**: While the rate looks standard, the truncation threshold τ must be carefully tuned based on the tail index ϵ.
- **Curse of Dimensionality**: As n_j increases, the estimation of each mode becomes more accurate due to the "blessing of dimensionality" in tensor structures.

Why 2 Iterations? — Asymptotic Normality

The paper provides a refined analysis of the iteration process $\iota = 0 \to 1 \to 2$:

- **Initialization ($\iota = 0$)**: Direct SVD on truncated data. It is **consistent** but not efficient; it doesn't fully exploit the tensor structure.
- **First Iteration ($\iota = 1$)**: The projection step reduces the noise from $O_p(1/\sqrt{n_k})$ to a much smaller order. It achieves the **optimal convergence rate**.
- **Second Iteration ($\iota = 2$)**: Crucial for **Statistical Inference**. The authors prove that after the 2nd projection, the error term:

$$\sqrt{T} \cdot \mathrm{vec}(\widehat{\mathbf{A}}_k^{(2)} - \mathbf{A}_k \mathbf{O}_k) \xrightarrow{d} \mathcal{N}(0, \mathbf{V}_k) \tag{8}$$

Key Takeaway

Two iterations are the "magic number": they balance **computational efficiency** (only 2 steps) with **inferential power** (enabling confidence intervals and p-values).

Methodology: Algorithm 2

Algorithm 2: Iterative Rank Estimation

Goal: Estimate $\mathbf{r} = (r_1, \ldots, r_d)$. **Criterion:** Eigenvalue Ratio (ER).

1. Initialization
Set $\widehat{r}_k^{(0)} = \bar{r}_k$. Obtain $\widehat{\mathbf{A}}_k^{(0)}$ using Algorithm 1.

2. Iterative Rank Update (for $\iota = 1, \ldots, N_{rank}$)

a) Compute $\tilde{\mathbf{\Sigma}}_k$ (Algorithm 1's projection using $\widehat{\mathbf{A}}_{j \neq k}^{(\iota-1)}$).

b) Update $\widehat{r}_k^{(\iota)}$ by maximizing the ER:

$$\widehat{r}_k^{(\iota)} = \text{arg max}_{1 \leq j \leq \bar{r}_k} \frac{\lambda_j(\tilde{\mathbf{\Sigma}}_k)}{\lambda_{j+1}(\tilde{\mathbf{\Sigma}}_k) + \delta_{n,T}}$$

c) Re-estimate $\widehat{\mathbf{A}}_k^{(\iota)}$ using the updated rank $\widehat{r}_k^{(\iota)}$.

3. Termination
Stop when $\widehat{r}_k^{(\iota)} = \widehat{r}_k^{(\iota-1)}$ or ι reaches the limit.

- **Consistency:** The paper proves $\mathbb{P}(\widehat{r}_k = r_k) \to 1$ as $n, T \to \infty$.

Theoretical Analysis: Main Theorem

Theorem. Consistency and Asymptotic Normality

Assume the noise $e_{t,\mathbf{i}}$ has finite $(2+2\epsilon)$-th moments for some $\epsilon \in (0,1]$, the factors are pervasive, and the truncation level satisfies $\tau \asymp (nT)^{1/(2+2\epsilon)}$.

Then as $\min(n_1, \ldots, n_d, T) \to \infty$, for each mode $k \in \{1, \ldots, d\}$, the TIP estimator $\widehat{\mathbf{A}}_k$ satisfies:

1. **Convergence Rate:**

$$d(\widehat{\mathbf{A}}_k, \mathbf{A}_k) \ = \ O_p\left(\frac{1}{\sqrt{T}} + \frac{1}{\sqrt{n^{(k)}}}\right), \quad \text{where } n^{(k)} = \prod_{j \neq k} n_j.$$

2. **Asymptotic Normality:** If the number of iterations $N \geq 2$,

$$\sqrt{T} \cdot \mathrm{vec}(\widehat{\mathbf{A}}_k - \mathbf{A}_k \mathbf{O}_k) \ \xrightarrow{d} \ \mathcal{N}(0, \mathbf{V}_k).$$

Numerical Simulations: Setup

We evaluate the **TIP (Trunc)** estimator against four comprehensive benchmarks:

- **iPE / TOPUP**: The iterative projection estimator by [1]. It is optimal under Gaussian noise but non-robust.
- **noTrunc**: An ablation version of TIP **without** the truncation step.
- **RTFA**: Robust Tensor Factor Analysis, a median-based or rank-based robust alternative.
- **PreAve**: A classical pre-averaging method used to mitigate high-dimensional noise.

Data Generation $(n_1, n_2, n_3 = 20, 30, 40)$

The noise $\mathcal{E}_t = \{e_{t,\mathbf{i}}\}$ is generated under two main settings:

1. **'no' (Gaussian)**: $e_{t,\mathbf{i}} \sim \mathcal{N}(0,1)$ to test efficiency loss.
2. **'idio' (Outliers)**: 0.5% of entries are replaced by large outliers (Cauchy-type) to test breakdown robustness.

Simulation Results: Robustness to Outliers

We present the finite-sample performance of five methods under different noise scenarios and outliers. The y-axis is in **log-scale** for better visualization of error gaps.

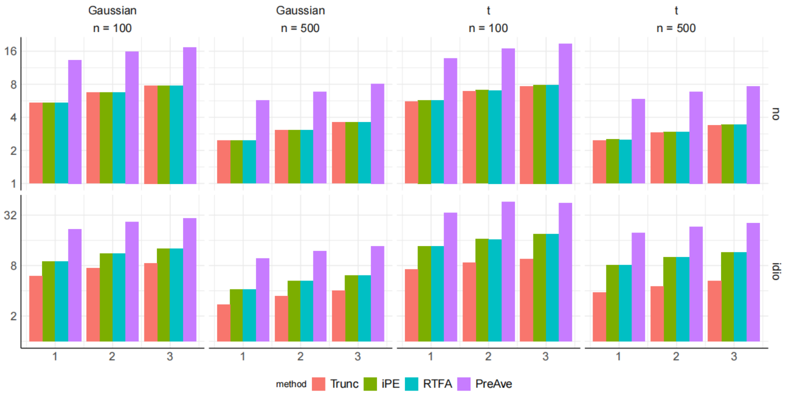

Figure: Average errors across scenarios.

Numerical Evidence: Asymptotic Normality

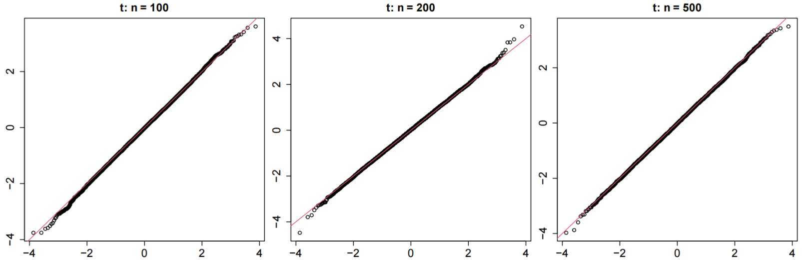

Figure 2: (T3): Plots of the sample quantiles of the scaled and centered entries of $\breve{\boldsymbol{\Lambda}}_k^{[2]}(\tau)$ (y-axis) against the quantiles from the standard normal distribution (x-axis) over varying $n \in \{100, 200, 500\}$ (left to right) when the data are generated from the t_3 distribution. In each plot, the $y = x$ line is given in red. See Appendix C.1.4 for full details.

Empirical Application: Euro Area Macro Data

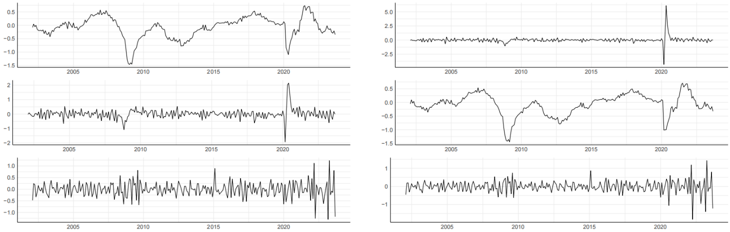

Figure 3: EA-MD: Factor time series $\widehat{f}_{1j,t}(\tau, \kappa)$ for $j = 1, 2, 3$ (top to bottom) with (left) and without (right) the truncation.

Takeaway: Truncation preserves the global factor structure against unprecedented economic shocks.

References I

[1] Wang, D., Liu, X., and Chen, R. (2021). Factor models for high-dimensional tensor time series. *Journal of the American Statistical Association*, 116(536), 1843-1859.