High-Dimensional Low-Rank Tensor Autoregressive Time Series Modeling

Presenter: Zhi-Long Han

December 29, 2025

Reference: Di Wang, Yao Zheng, Guodong Li, *High-Dimensional Low-Rank Tensor Autoregressive Time Series Modeling*, arXiv:2101.04276, 2021.

Outline

Why Tensor-Valued Time Series?

Tensor-valued observation: at each time t we observe

$$Y_t \in \mathbb{R}^{p_1 \times p_2 \times \cdots \times p_d}, \qquad E_t \in \mathbb{R}^{p_1 \times \cdots \times p_d}.$$

- Examples: portfolio panels, multi-way traffic flows, dynamic images/videos.
- Tensor keeps multi-way structure that vectorization destroys.

Naïve baseline: vectorize then VAR(1)

$$y_t = Ay_{t-1} + e_t, \quad y_t = \mathrm{vec}(Y_t) \in \mathbb{R}^p,\ p = \prod_{i=1}^{d} p_i,\ A \in \mathbb{R}^{p \times p}.$$

- Parameter explosion: p^2 entries in $A \Rightarrow$ infeasible when p is large and T is limited.
- Interpretability loss: no longer see effects along each mode.

Motivation: What Structure Can We Exploit?

Key Insight

Replace the huge VAR matrix $A \in \mathbb{R}^{p \times p}$ by a **transition tensor**

$$\mathcal{A} \in \mathbb{R}^{p_1 \times \cdots \times p_d \times p_1 \times \cdots \times p_d}$$

with **low Tucker multilinear ranks** $\boldsymbol{r} = (r_1, \ldots, r_{2d})$, where $r_k \ll p_k$.

What this paper delivers

- LRTAR model + dynamic factor interpretation via Tucker.
- Two estimation regimes: LTR (asymptotic) and SSN/TSSN (high-dimensional).
- New SSN regularizer (all square matricizations) + rank selection consistency.

LRTAR(1): Model and VAR View

Tensor autoregression:

$$Y_t = \langle \mathcal{A},\, Y_{t-1} \rangle + E_t, \tag{1}$$

where

- $Y_t, E_t \in \mathbb{R}^{p_1 \times \cdots \times p_d}$,
- $\mathcal{A} \in \mathbb{R}^{p_1 \times \cdots \times p_d \times p_1 \times \cdots \times p_d}$ is a $2d$-th order transition tensor,
- contraction over the last d modes:

$$\big(\langle \mathcal{A}, Y_{t-1} \rangle\big)_{i_1,\ldots,i_d} = \sum_{j_1=1}^{p_1} \cdots \sum_{j_d=1}^{p_d} \mathcal{A}_{i_1,\ldots,i_d,j_1,\ldots,j_d} \, (Y_{t-1})_{j_1,\ldots,j_d}.$$

VAR view (square matricization): let $p = \prod_{i=1}^{d} p_i$, $y_t = \mathrm{vec}(Y_t) \in \mathbb{R}^p$. Define $\mathcal{A}_{[S_2]} \in \mathbb{R}^{p \times p}$ by grouping $(i_1, \ldots, i_d)$ as rows and $(j_1, \ldots, j_d)$ as columns. Then

$$y_t = \mathcal{A}_{[S_2]} y_{t-1} + e_t, \quad e_t = \mathrm{vec}(E_t),$$

where $\rho(\mathcal{A}_{[S_2]}) < 1$.

Low-Rank Transition Tensor

Low-rank structure assumption (Tucker):

$$\mathcal{A} = \mathcal{G} \times_1 U_1 \times_2 U_2 \cdots \times_{2d} U_{2d}, \tag{2}$$

where

- $\mathcal{A} \in \mathbb{R}^{p_1 \times \cdots \times p_d \times p_1 \times \cdots \times p_d}$
- Core tensor: $\mathcal{G} \in \mathbb{R}^{r_1 \times \cdots \times r_{2d}}$
- Factor matrices: $U_k \in \mathbb{R}^{p_k \times r_k}$ $(k = 1, \ldots, d)$.

Why Tucker helps (parameter reduction):

Model	Number of parameters (order)
Naïve VAR	p^2 $(p = \prod_{k=1}^{d} p_k)$
LRTAR with Tucker	$\prod_{k=1}^{2d} r_k \ + \ \sum_{k=1}^{d} p_k r_k \ + \ \sum_{k=1}^{d} p_k r_{d+k}$

Dynamic Factor Interpretation

low-dimensional dynamics

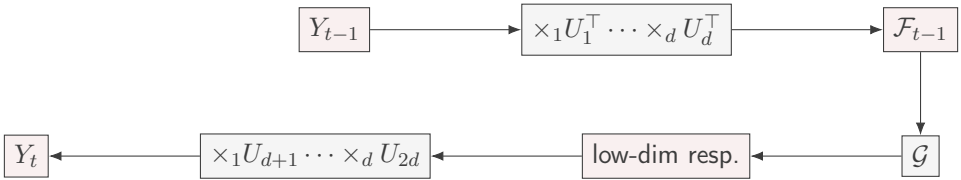
Let $\mathcal{F}_{t-1} = Y_{t-1} \times_1 U_1^\top \times_2 \cdots \times_d U_d^\top \in \mathbb{R}^{r_1 \times \cdots \times r_d}$. Then the response-side projection satisfies

$$Y_t \times_1 U_{d+1}^\top \times_2 \cdots \times_d U_{2d}^\top \;=\; \langle \mathcal{G},\, \mathcal{F}_{t-1} \rangle \;+\; E_t \times_1 U_{d+1}^\top \times_2 \cdots \times_d U_{2d}^\top. \qquad (3)$$

Hence, (3) is a low-dim tensor regression defined on the projections of Y_t and Y_{t-1}.

Interpretation

- $U_1, \ldots, U_d$: predictor loadings (compress Y_{t-1}).
- $U_{d+1}, \ldots, U_{2d}$: response loadings (compress Y_t).
- $\mathcal{G}$ links low-dim predictors to low-dim responses.



Two Estimation Regimes

Regime A: Low-dimensional (exact low-rank)

- Tucker ranks $\boldsymbol{r}$ are known; T is large relative to effective dimension.
- Estimate by **rank-constrained least squares**.
- Result: asymptotic normality (Theorem 1).

Regime B: High-dimensional (approximate low-rank)

- $p = \prod p_i$ large; $\mathcal{A}$ is only *approximately* low-rank.
- Estimate by convex regularization:
  - SN: sum of nuclear norms of one-mode matricizations.
  - MN: nuclear norm of a single $p \times p$ square matricization.
  - SSN (proposed): sum of nuclear norms of *all* square matricizations.
- Result: non-asymptotic error bounds + rank selection (Theorems 2–5).

Low-Dimensional Estimator: low-Tucker-rank (LTR)

Objective (LTR):

$$\widehat{\mathcal{A}}_{\mathrm{LTR}} = \arg\min_{\mathcal{A}} \frac{1}{T-1} \sum_{t=2}^{T} \left\| Y_t - \langle \mathcal{A},\, Y_{t-1} \rangle \right\|_F^2 \tag{4}$$

$$\text{s.t.} \quad \mathrm{rank}_k(\mathcal{A}) \le r_k, \qquad k = 1, \ldots, 2d.$$

Unknowns and sizes:

- Transition tensor: $\mathcal{A} \in \mathbb{R}^{p_1 \times \cdots \times p_d \times p_1 \times \cdots \times p_d}$.
- Tucker factors and core:

$$U_k \in \mathbb{R}^{p_k \times r_k} \ (k = 1, \ldots, d), \qquad U_{d+k} \in \mathbb{R}^{p_k \times r_{d+k}} \ (k = 1, \ldots, d),$$

$$\mathcal{G} \in \mathbb{R}^{r_1 \times \cdots \times r_{2d}}, \qquad \mathcal{A} = \mathcal{G} \times_1 U_1 \cdots \times_{2d} U_{2d}.$$

Computation

Solve (4) via **ALS** (alternating LS / Procrustes updates for U_k and $\mathcal{G}$).

Algorithm 1: ALS for LTR

Algorithm: Alternating Least Squares (ALS)

Inputs	$\{Y_t\}_{t=1}^{T}$, ranks $\boldsymbol{r} = (r_1, \ldots, r_{2d})$, max iters S, tolerance ε
Outputs	$\widehat{\mathcal{A}}_{\mathrm{LTR}}$ (or $\widehat{\mathcal{G}}, \{\widehat{U}_k\}_{k=1}^{2d}$)
Init	Compute $\mathcal{A}^{(0)}$ (e.g., OLS/RRR), then HOSVD to obtain $\mathcal{G}^{(0)}, \{U_k^{(0)}\}_{k=1}^{2d}$
Updating	(1) Update predictor loadings $U_k^{(s+1)} \in \mathbb{R}^{p_k \times r_k}$, $k = 1, \ldots, d$ (*Procrustes*) (2) Update response loadings $U_{d+k}^{(s+1)} \in \mathbb{R}^{p_k \times r_{d+k}}$, $k = 1, \ldots, d$ (*Procrustes*) (3) Update core $\mathcal{G}^{(s+1)} \in \mathbb{R}^{r_1 \times \cdots \times r_{2d}}$ (*LS*) (4) Form $\mathcal{A}^{(s+1)} = \mathcal{G}^{(s+1)} \times_1 U_1^{(s+1)} \cdots \times_{2d} U_{2d}^{(s+1)}$
Stop	If $\|\mathcal{A}^{(s+1)} - \mathcal{A}^{(s)}\|_F / \|\mathcal{A}^{(s)}\|_F \le \varepsilon$

Theorem 1: Asymptotic Normality of LTR (Low-dimensional)

Theorem 1

Let $\widehat{\mathcal{A}}_{LTR}$ be the rank-constrained LS estimator. Assume $\mathbb{E}\|e_t\|_2^2 < \infty$ and stationarity holds. Then

$$\sqrt{T}\,\mathrm{vec}\big((\widehat{\mathcal{A}}_{LTR} - \mathcal{A})_{[S_2]}\big) \;\Rightarrow\; \mathcal{N}\big(0, \Sigma_{LTR}\big),$$

where $(\cdot)_{[S_2]}$ is the square matricization and $\Sigma_{LTR} = H\,(H^\top J H)^\dagger H^\top \in \mathbb{R}^{p^2 \times p^2}$.

Interpretation:
If the Tucker ranks are correctly specified, LTR is a **well-specified** low-dimensional estimator; under Gaussian noise, least squares coincides with the **Maximum Likelihood Estimator (MLE)**, yielding the classical $\mathcal{O}(T^{-1/2})$ estimation rate.

High-Dimensional Regularization: Unified Template

When p is large, estimate $\mathcal{A}$ via:

$$\widehat{\mathcal{A}} = \arg\min_{\mathcal{A}} \frac{1}{T-1} \sum_{t=2}^{T} \|Y_t - \langle \mathcal{A}, Y_{t-1} \rangle\|_F^2 + \lambda \mathcal{R}(\mathcal{A}). \tag{5}$$

Approximate low-rank matrix class For a matrix $M \in \mathbb{R}^{m \times n}$ with singular values $\sigma_j(M)$, define

$$\mathbb{B}_q(r_q; m, n) = \Big\{ M : \sum_{j \geq 1} \sigma_j(M)^q \leq r_q,\ 0 \leq q < 1 \Big\}.$$

- $q = 0$: exactly rank-r_0 matrices.
- $0 < q < 1$: singular values decay (nearly low-rank).

SN vs MN vs SSN: Regularizers

Abbrev.: SN: Sum of Nuclear norms; MN: Matrix Nuclear norm; SSN: Sum of Square Nuclear norms.

Notation: $p_{-i} = \prod_{j=1, j\neq i}^{d} p_j, \quad p = \prod_{j=1}^{d} p_j.$

Method	Matrix shape (how obtained + size)	Regularizer
SN	Matricize along mode i, $i = 1, \ldots, 2d$ $\mathcal{A}_{(i)} \in \mathbb{R}^{p_i \times (p_{-i} p)}$	$\mathcal{R}_{SN}(\mathcal{A}) = \sum_{i=1}^{2d} \|\mathcal{A}_{(i)}\|_*$
MN	VAR view; one fixed square view S_1 $\mathcal{A}_{[S_1]} \in \mathbb{R}^{p \times p}$	$\mathcal{R}_{MN}(\mathcal{A}) = \|\mathcal{A}_{[S_1]}\|_*$
SSN	All square views I_k, $k = 1, \ldots, 2^{d-1}$ $\mathcal{A}_{[I_k]} \in \mathbb{R}^{p \times p}$	$\mathcal{R}_{SSN}(\mathcal{A}) = \sum_{k=1}^{2^{d-1}} \|\mathcal{A}_{[I_k]}\|_*$

Takeaway

SSN keeps **square-matrix structure** (like MN) but enforces it across *all* square views.

Algorithm 2: ADMM for SSN

To handle multiple nuclear norms in $\|\mathcal{A}\|_{SSN}$, introduce auxiliary variables $W_k \in \mathbb{R}^{p_1 \times \cdots \times p_d \times p_1 \times \cdots \times p_d} (k = 1, \ldots, 2^{d-1})$ with constraints $W_k = \mathcal{A}$.

Augmented Lagrangian Function

$$\mathcal{L}(\mathcal{A}, \{W_k\}, \{C_k\}) = L_T(\mathcal{A}) + \sum_{k=1}^{2^{d-1}} \Big(\lambda_{SSN} \|(W_k)_{[I_k]}\|_* + \frac{\rho}{2} \|\mathcal{A} - W_k + C_k\|_F^2 \Big).$$

ADMM iterations (Algorithm 2)

- **A-update:** $\mathcal{A}^{(j+1)} = \arg\min_{\mathcal{A}} \; L_T(\mathcal{A}) + \frac{\rho}{2} \sum_k \|\mathcal{A} - W_k^{(j)} + C_k^{(j)}\|_F^2$.
- **W-update (parallel over k):**

$$(W_k^{(j+1)})_{[I_k]} = \mathrm{SVT}_{\lambda_{SSN}/\rho} \Big((\mathcal{A}^{(j+1)} + C_k^{(j)})_{[I_k]} \Big),$$

  where $\mathrm{SVT}_\tau(\cdot)$ soft-thresholds singular values by τ.
- **Dual:** $C_k^{(j+1)} = C_k^{(j)} + \mathcal{A}^{(j+1)} - W_k^{(j+1)}$.

Assumptions and Key Constants

High-level assumptions

- Stationary VAR(1) representation (spectral radius < 1).
- Innovations are sub-Gaussian with parameter κ (controls tails).
- Restricted strong convexity (RSC): loss $L_T(\mathcal{A})$ behaves like a quadratic over low-rank tangent cones.

Notation used in bounds Let $\Sigma_e = \mathrm{Var}(e_t) \in \mathbb{R}^{p \times p}$, and define constants

$$M_1 = \frac{\lambda_{\max}(\Sigma_e)}{\mu_{\min}(\mathcal{A})^{1/2}}, \qquad M_2 = \frac{\lambda_{\min}(\Sigma_e)\,\mu_{\max}(\mathcal{A})}{\lambda_{\max}(\Sigma_e)\,\mu_{\min}(\mathcal{A})},$$

where $\mu_{\min}, \mu_{\max}$ capture stability/conditioning of the process.

Role

Bad stability / ill-conditioning $\Rightarrow$ small M_2 $\Rightarrow$ larger sample size needed.

Theorem 2: SN Estimator (Non-asymptotic Error Bound)

SN estimator

$$\widehat{\mathcal{A}}_{SN} = \arg\min_{\mathcal{A}} L_T(\mathcal{A}) + \lambda_{SN} \sum_{i=1}^{2d} \|\mathcal{A}_{(i)}\|_*.$$

Theorem 2 (core condition + bound)

If $T \gtrsim \max_{1\le i\le d}(p_{-i}p) \;+\; \max(\kappa^2, \kappa^4) M_2^{-2}\, p,$, $\lambda_{SN} \gtrsim \kappa^2 M_1 d^{-2} \sum_{i=1}^{2d} \sqrt{\frac{p_{-i}p}{T}}$, and $\mathcal{A}_{(i)} \in \mathbb{B}_q(r_q^{(i)}; p_i, p_{-i}p)$ for $0 \le q < 1$, then w.h.p.

$$\|\widehat{\mathcal{A}}_{SN} - \mathcal{A}\|_F \;\lesssim\; \sqrt{r_q}\Big(\frac{2d\,\lambda_{SN}}{\alpha_{RSC}}\Big)^{1-q/2}, \quad r_q = \frac{1}{2d}\sum_{i=1}^{2d} r_q^{(i)}.$$

Why worse: $\mathcal{A}_{(i)} \in \mathbb{R}^{p_i \times (p_{-i}p)}$ is highly unbalanced $\Rightarrow$ deviation dominated by $p_{-i}p$.

Theorem 3–4: MN vs SSN

Theorem 3 (MN) and Theorem 4 (SSN): bounds

MN: if $T \gtrsim [1 + \max(\kappa^2, \kappa^4) M_2^{-2}]\, p$ and $\lambda_{MN} \gtrsim \kappa^2 M_1 \sqrt{p/T}$, with $\mathcal{A}_{[S_1]} \in \mathbb{B}_q(s_q^{(1)}; p, p)$,

$$\|\widehat{\mathcal{A}}_{MN} - \mathcal{A}\|_F \;\lesssim\; \sqrt{s_q^{(1)}} \Big(\frac{\lambda_{MN}}{\alpha_{RSC}} \Big)^{1-q/2}.$$

SSN: if $T \gtrsim [1 + \max(\kappa^2, \kappa^4) M_2^{-2}]\, p$ and $\lambda_{SSN} \gtrsim \kappa^2 M_2^{1-d} \sqrt{p/T}$, with $\mathcal{A}_{[I_k]} \in \mathbb{B}_q(s_q^{(k)}; p, p)$,

$$\|\widehat{\mathcal{A}}_{SSN} - \mathcal{A}\|_F \;\lesssim\; \sqrt{s_q} \Big(\frac{2^{d-1} \lambda_{SSN}}{\alpha_{RSC}} \Big)^{1-q/2}, \quad s_q = 2^{1-d} \sum_{k=1}^{2^{d-1}} s_q^{(k)}.$$

Sample Size & Error Rate Comparison

Define $p_{-i} = \prod_{j \neq i} p_j$, $p = \prod_{j=1}^{d} p_j$.

Estimator	Sample size condition (order)	Frobenius error rate (order)
SN	$T \gtrsim \max_{i \leq d}(p_{-i}p) + M_2^{-2}p$	$\sqrt{r_q}\left(\max_{i \leq d} \frac{p_{-i}p}{T}\right)^{\frac{1}{2}-\frac{q}{4}}$
MN	$T \gtrsim (1 + M_2^{-2})p$	$\sqrt{s_q^{(1)}}\left(\frac{p}{T}\right)^{\frac{1}{2}-\frac{q}{4}}$
SSN	$T \gtrsim (1 + M_2^{-2})p$	$\sqrt{s_q}\left(\frac{p}{T}\right)^{\frac{1}{2}-\frac{q}{4}}$

How to read this table

- SN is slower because $p_{-i}p$ can be much larger than p (unbalanced unfoldings).
- MN improves by using a **single** $p \times p$ square view.
- **SSN** keeps the same p/T scaling but is more robust by enforcing low-rank across *all* square views.

Theorem 5: TSSN Rank Selection Consistency

Goal: consistently recover the true Tucker ranks $\{\mathrm{rank}(\mathcal{A}_{(i)})\}_{i=1}^{2d}$. **Mechanism:**

Truncated SSN (TSSN)

- Obtain $\widehat{\mathcal{A}}_{SSN}$ first, then perform Hard Thresholding on singular values of each unfolding $(\widehat{\mathcal{A}}_{SSN})_{(i)}$.
- Keep singular values $\sigma_j \geq \gamma$ (typically $\gamma \propto \lambda_{SSN}$), discard the rest.

Theorem 5 (statement)

Under the conditions of Theorem 4 and an additional signal strength assumption (Assumption 3), if $\lambda_{SSN} \asymp \kappa^2 M_2^{1-d}\sqrt{p/T}$, then as $T \to \infty$ (fixed d),

$$\mathbb{P}\Big(\mathrm{rank}\big((\widehat{\mathcal{A}}_{TSSN})_{(i)}\big) = \mathrm{rank}\big(\mathcal{A}_{(i)}\big),\ i = 1, \ldots, 2d\Big) \to 1,$$

and

$$\|\widehat{\mathcal{A}}_{TSSN} - \mathcal{A}\|_F = \mathcal{O}_p\Big(\sqrt{s_0\,\frac{p}{T}}\Big),$$

where s_0 is the exact low-rank complexity parameter in Assumption 3.

Experimental Design: Two-Track Validation

Track 1: Simulation Studies (Validation)

- **Goal:** Verify estimation accuracy and validate theoretical scaling rates.
- **Scope:** Low-Dim (LTR vs RRR) & High-Dim (SN vs MN vs SSN).
- **Metric:** Tensor estimation error $\|\widehat{\mathcal{A}} - \mathcal{A}\|_F$.

Track 2: Real Data Analysis (Application)

- **Goal:** Test predictive power and interpret learned low-rank structures.
- **Datasets:** 10×10 Matrix Portfolios ($d = 2$) & $4 \times 4 \times 2$ Tensor Portfolios ($d = 3$).
- **Metric:** Rolling forecasting error (RMSFE).

Benchmarking: Baselines & Estimators

1. **Existing Baseline Methods**
   - **Vector-based:** VAR (Standard Vector AR), VFM (Vector Factor Model).
   - **Tensor-based:** MAR/MTAR (Separable), MFM/TFM (Factor models).

2. Proposed LRTAR Estimators (Ours)

Low-Dimensional Regime	High-Dimensional Regime
LTR (Rank-constrained LS)	**SN** / **MN** (Standard norms) **SSN** (Sum of Square-matrix Nuclear norms) **TSSN** (SSN + Truncation)

Simulation Setup & Protocols

1. Data Generation (Controlled Environment)

- **Process:** Generate stationary tensor series from LRTAR(1):

$$\mathcal{Y}_t = \langle \mathcal{A}, \mathcal{Y}_{t-1} \rangle + \mathcal{E}_t, \quad \mathcal{E}_t \sim \mathcal{N}(0, I).$$

- **Variables:** Sample size T, Dimension $p = \prod p_i$, Ranks r, Noise levels.

2. Estimators Comparison Groups

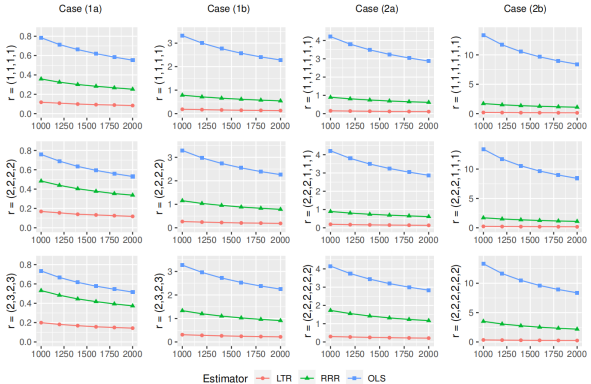
- **Low-Dimensional Regime:**
  LTR (Ours) vs. RRR vs. OLS — *Test efficiency gain from tensor structure.*
- **High-Dimensional Regime:**
  SN vs. MN vs. SSN vs. TSSN — *Test benefits of square matricization.*

Simulation I: Low-Dimensional Estimation Efficiency

Setup: Low-rank LRTAR model ($d = 2, 3$) with varying dimensions p_i and ranks r.

Key Finding

The proposed **LTR estimator** achieves the lowest estimation error by strictly enforcing tensor rank constraints, significantly outperforming RRR and OLS.

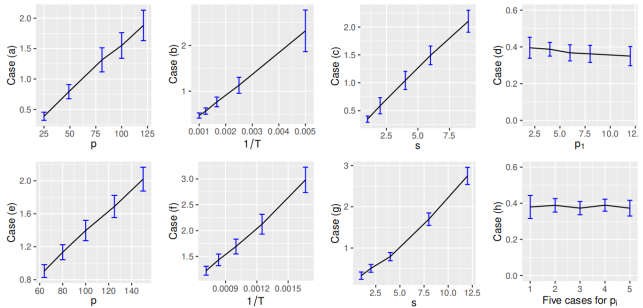


Simulation II: Verification of Theoretical Rates (High-Dim)

Setup: Verify error bound for SSN estimator. Varying p, T, and complexity s_0.

Key Finding

The estimation error scales **linearly** with total dimension p and complexity s_0, confirming that SSN is robust and effective for high-dimensional scaling.

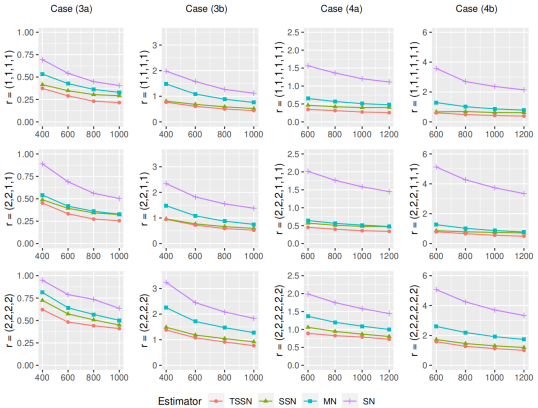


Simulation III: Comparison of High-Dimensional Estimators

Comparison: SN vs. MN vs. SSN vs. TSSN.

Key Finding

TSSN yields the lowest error by combining the efficiency of balanced square matricizations (SSN) with rank truncation, significantly beating SN and MN.



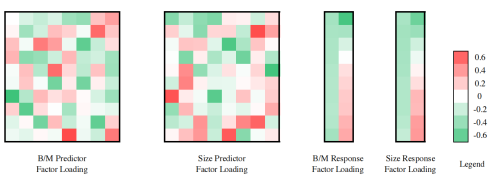
Real Data: Fama-French Portfolios Analysis

Data: Monthly returns of 10×10 portfolios ($d = 2$) & $4 \times 4 \times 2$ portfolios ($d = 3$).
Metric: Out-of-sample rolling forecast error.

Key Results

- **Forecasting:** LRTAR (TSSN) achieves the lowest out-of-sample error.
- **Structure:** Identified factors show *Market* (uniform) and *Size/Value* (monotonic) patterns.

	Model	VAR	VFM	MAR	MFM	LRTAR		Best	Worst
						SSN	TSSN		
In-sample	ℓ_2 norm	**31.61**	35.85	34.12	35.86	33.18	33.38	VAR	MFM
	ℓ_0 norm	**8.09**	9.23	9.12	9.24	8.84	8.89	VAR	MFM
Out-of-sample	ℓ_2 norm	39.84	39.11	35.26	38.53	32.60	**31.47**	TSSN	VAR
	ℓ_∞ norm	11.98	12.30	11.47	11.00	9.53	**9.33**	TSSN	VFM



Experimental Summary & Implications

Simulation Insights

- **Low-Dim:** LTR beats vector baselines when ranks are specified correctly.
- **High-Dim:**
  - **SN** suffers from unbalanced (rectangular) matricizations.
  - **MN/SSN** gain efficiency via balanced square views.
  - **TSSN** further improves accuracy via rank truncation.

Real Data Evidence

- **Forecasting:** SSN/TSSN achieve the best out-of-sample performance.
- **Interpretability:**
  - Response-side low ranks suggest a *low-dimensional predictable subspace*.
  - Factors align with financial theory (Market/Size effects).

Conclusion

Structured low-rank dynamics (LRTAR) significantly improve both **prediction accuracy** and **model interpretability**.

Main Takeaways

1. **The Methodology: LRTAR**

$$\mathcal{Y}_t = \langle \mathcal{A}, \mathcal{Y}_{t-1} \rangle + \mathcal{E}_t, \quad \mathcal{A} \text{ has low Tucker-rank.}$$

2. **Key Contributions**
   - **Two Regimes:** Asymptotic normality (Low-dim) vs. Finite-sample rates (High-dim).
   - **SSN Regularizer:** The key innovation. Exploits **balanced square matricizations** to beat standard SN.
   - **TSSN Estimator:** Provides rank recovery with theoretical consistency.

Limitations & Future Directions

- Currently limited to AR(1); extending to AR(p) requires stability selection.
- Computational cost: SSN-ADMM involves 2^{d-1} SVDs (heavy computation when total dimension p is very large).

References

- (Wang et al.'24) "High-dimensional low-rank tensor autoregressive time series modeling." *Journal of Econometrics*, 2024, 238(1): 105544.
- (Wang et al.'22) "High-dimensional vector autoregressive time series modeling via tensor decomposition." *Journal of the American Statistical Association*, 2022, 117(539): 1338–1356.
- (Mu et al.'14) "Square Deal: Lower Bounds and Improved Relaxations for Tensor Recovery." *ICML*, 2014.
- (Negahban & Wainwright'11) "Estimation of (near) low-rank matrices with noise and high-dimensional scaling." *Annals of Statistics*, 2011.
- (Lam et al.'12) "Factor modeling for high-dimensional time series: inference for the number of factors." *Annals of Statistics*, 2012.