

Markov Switching Multiple-equation Tensor Regression

Roberto Casarin[‡], Radu Craiu[†], **Qing Wang[‡]**

[†]University of Toronto

[‡]**Ca' Foscari University of Venice**

25th June 2025

IWFOS 2025, Novara, Italy

Data Structure

As data grow in volume and complexity, it is increasingly common to record them as high-dimensional arrays or tensors, in many fields, such as neuroimaging (Spencer et al., 2022; Guha and Rodriguez, 2021), biostatistics, financial networks (Billio et al., 2024), or even more generally, in time series (Billio et al., 2023).

Motivation

Data Structure

As data grow in volume and complexity, it is increasingly common to record them as high-dimensional arrays or tensors, in many fields, such as neuroimaging (Spencer et al., 2022; Guha and Rodriguez, 2021), biostatistics, financial networks (Billio et al., 2024), or even more generally, in time series (Billio et al., 2023).

Tensor Data Regressions

Scalar response on covariate tensor (Guha and Rodriguez, 2021) or tensor response on covariate tensor (Wang and Xu, 2024; Billio et al., 2023).

Motivation

Data Structure

As data grow in volume and complexity, it is increasingly common to record them as high-dimensional arrays or tensors, in many fields, such as neuroimaging (Spencer et al., 2022; Guha and Rodriguez, 2021), biostatistics, financial networks (Billio et al., 2024), or even more generally, in time series (Billio et al., 2023).

Tensor Data Regressions

Scalar response on covariate tensor (Guha and Rodriguez, 2021) or tensor response on covariate tensor (Wang and Xu, 2024; Billio et al., 2023).

Modelling Issues

Tensor models have been studied extensively in a linear framework. Common challenge: model misspecification, [regimes](#), breaks,.... Solution: time-varying parameters.

A Tensor Regression Model

A new Bayesian **tensor model for multiple-equation regressions** that accounts for latent regime changes is proposed.

- Ⓐ We extend the **soft tensor** linear regression models (Guha and Rodriguez, 2021; Papadogeorgou et al., 2021; Wang and Xu, 2024) to an HMM (or MS) framework to accommodate structural breaks.

A Tensor Regression Model

A new Bayesian **tensor model for multiple-equation regressions** that accounts for latent regime changes is proposed.

- A We extend the **soft tensor** linear regression models (Guha and Rodriguez, 2021; Papadogeorgou et al., 2021; Wang and Xu, 2024) to an HMM (or MS) framework to accommodate structural breaks.
- B The **Parallel Factor (PARAFAC)** representation, also called CANDECOMP/PARAFAC or Polyadic Decomposition (see Kolda and Bader, 2009), of the coefficient tensor is driven by a common **hidden Markov chain** process.

A Tensor Regression Model

A new Bayesian **tensor model for multiple-equation regressions** that accounts for latent regime changes is proposed.

- Ⓐ We extend the **soft tensor** linear regression models (Guha and Rodriguez, 2021; Papadogeorgou et al., 2021; Wang and Xu, 2024) to an HMM (or MS) framework to accommodate structural breaks.
- Ⓑ The **Parallel Factor (PARAFAC)** representation, also called CANDECOMP/PARAFAC or Polyadic Decomposition (see Kolda and Bader, 2009), of the coefficient tensor is driven by a common **hidden Markov chain** process.
- Ⓒ We consider a **multi-equation** setting with possibly different response variables across equations.

A Tensor Regression Model

A new Bayesian **tensor model for multiple-equation regressions** that accounts for latent regime changes is proposed.

- Ⓐ We extend the **soft tensor** linear regression models (Guha and Rodriguez, 2021; Papadogeorgou et al., 2021; Wang and Xu, 2024) to an HMM (or MS) framework to accommodate structural breaks.
- Ⓑ The **Parallel Factor (PARAFAC)** representation, also called CANDECOMP/PARAFAC or Polyadic Decomposition (see Kolda and Bader, 2009), of the coefficient tensor is driven by a common **hidden Markov chain** process.
- Ⓒ We consider a **multi-equation** setting with possibly different response variables across equations.
- Ⓓ A **Bayesian inference** procedure that relies on numerical exploration of the posterior via a new and efficient Gibbs sampler, which reduces computational costs and improves scalability.

Tensor Algebra

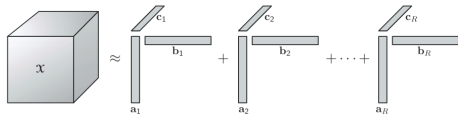
Tensor Representation

Two major tensor representation methods often used in the literature: CP (CANDECOMP/PARAFAC) and Tucker. (Kolda and Bader (2009))

CP Representation: given a 3-mode tensor $\mathcal{B} \in \mathbb{R}^{I \times J \times K}$

$$\mathcal{B} = \sum_{d=1}^D \mathbf{a}_d \otimes \mathbf{b}_d \otimes \mathbf{c}_d$$

where $\mathbf{a}_d \in \mathbb{R}^I$, $\mathbf{b}_d \in \mathbb{R}^J$, $\mathbf{c}_d \in \mathbb{R}^K$ are the marginals from the CP decomposition, D is the rank of the tensor, \otimes represents the *outer* product.



Dimensionality reduction: $I \times J \times K \rightarrow (I + J + K)D$.

Tensor Algebra

Hard vs Soft PARAFAC

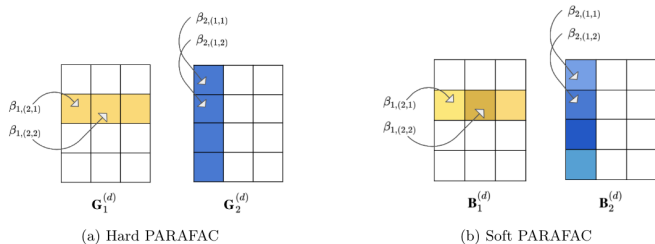


Figure: Hard vs Soft PARAFAC (Papadogeorgou et al., 2021)

$$\beta_{m,jm}^{(d)} \sim \mathcal{N}_{q_m} \left(\gamma_{m,jm}^{(d)}, \tau \sigma_m^2 \zeta^{(d)} I_{q_m} \right), q_m = \prod_{l \neq m} p_l$$

$$\mathbb{E} \left(B | \gamma_1^{(d)}, \gamma_2^{(d)} \right) = \sum_{d=1}^D \mathbb{E} \left(B_1^{(d)} \right) * \mathbb{E} \left(B_2^{(d)} \right) = \sum_{d=1}^D G_1^{(d)} * G_2^{(d)} = \sum_{d=1}^D \gamma_1^{(d)} \circ \gamma_2^{(d)}$$

The Model

A Markov-Switching Multiple-equation Tensor Regression Model:

$$\begin{cases} y_{1t} = \mu_1(s_t) + \langle B_1(s_t), X_t \rangle + \sigma_1(s_t)\varepsilon_{1t} \\ \vdots \\ y_{Nt} = \mu_N(s_t) + \langle B_N(s_t), X_t \rangle + \sigma_N(s_t)\varepsilon_{Nt} \end{cases} \quad (1)$$

where $t = 1, \dots, T$, $X_t, B_\ell(s_t)$ are $p_1 \times p_2$ matrices, $\langle \cdot, \cdot \rangle$ denotes inner product. The latent process is a K -state Markov chain process and the parametrization used is

$$\mu_\ell(s_t) = \sum_{k=1}^K \mu_{\ell k} \mathbb{I}(s_t = k), \quad B_\ell(s_t) = \sum_{k=1}^K B_{\ell k} \mathbb{I}(s_t = k), \quad \sigma_\ell(s_t) = \sum_{k=1}^K \sigma_{\ell k} \mathbb{I}(s_t = k)$$

Assume the following decomposition:

$$B_{\ell k} = \sum_{d=1}^D B_{\ell, k, 1}^{(d)} * B_{\ell, k, 2}^{(d)}$$

where $*$ is the *Hadamard product*, $B_{\ell, k, m}^{(d)}$, $m = 1, 2$ are the multiplicative factors. D is the number of components used to decompose the tensor.

Hierarchical Priors

We use shrinkage priors to favor sparsity:

$$B_m^{(d)} \sim \mathcal{MN}_{p_1, p_2} \left(G_m^{(d)}, \tau \sigma_m^2 \zeta^{(d)} I_{p_1}, I_{p_2} \right) \quad (2)$$

$$\gamma_m^{(d)} \sim \mathcal{N}_{p_m}(\mathbf{0}, \tau \zeta^{(d)} W_m^{(d)}) \quad (3)$$

$$w_{m, j_m}^{(d)} \sim \text{Exp}((\lambda_m^{(d)})^2 / 2) \quad (4)$$

$$\lambda_m^{(d)} \sim \mathcal{Ga}(a_\lambda, b_\lambda) \quad (5)$$

$$\sigma_m^2 \sim \mathcal{Ga}(a_\sigma, b_\sigma) \quad (6)$$

$$\tau \sim \mathcal{Ga}(a_\tau, b_\tau) \quad (7)$$

$$(\zeta^{(1)}, \dots, \zeta^{(D)}) \sim \text{Dir}(\alpha/D, \dots, \alpha/D) \quad (8)$$

where $m = \{1, 2\}$ is the number of mode, p_1, p_2 are the size of each mode.

$$G_m^{(d)} = \begin{cases} \gamma_1^{(d)} \circ \iota_{p_2} & m = 1 \\ \iota_{p_1} \circ \gamma_2^{(d)} & m = 2 \end{cases}$$

Selection of Hyperparameters

The choice of **hyperparameters** can have a large effect on the performance of the model. We follow the strategy in (Papadogeorgou et al. (2021)) to choose the hyperparameters by studying the properties of **induced prior variance** on the coefficients B .

In particular, we choose the hyperparameters such that $\text{Var}(B_{ij}) = V^*$ and the additional variance introduced by the softening equals to AV^* . B_{ij} denotes the entry of B .

We found that:

$$\mathbb{V}(B_{ij}) = \frac{a_\tau(a_\tau + 1)}{b_\tau^2} C \left(\frac{a_\sigma}{b_\sigma} + \frac{2b_\lambda^2}{(a_\lambda - 1)(a_\lambda - 2)} \right)^2 \quad (9)$$

$$\frac{a_\sigma}{b_\sigma} = \frac{b_\tau}{a_\tau} \sqrt{\frac{a_\tau V^*}{(a_\tau + 1)C}} \left(1 - \sqrt{1 - AV^*} \right) \quad (10)$$

where $C = \frac{\frac{\alpha}{D} + 1}{\alpha + 1}$.

In simulation we use $V^* = 1$ and $AV^* = 10\%$.

Full Conditionals

Let $\boldsymbol{\theta} = (\boldsymbol{\theta}_1, \dots, \boldsymbol{\theta}_K)$ be the collection of the **state-specific** parameters $\boldsymbol{\theta}_k = (\beta_k, \gamma_k, \zeta_k, \tau_k, \boldsymbol{\lambda}_k, \boldsymbol{w}_k, \sigma_k^2, \mu_k)$ and denote with $\mathbf{y} = (y_1, \dots, y_T)$, $\mathbf{X} = (X_1, \dots, X_T)$ and $\mathbf{s} = (s_1, \dots, s_T)$ the collection of **response variables**, **covariates** and **state variables**, respectively. The joint posterior of the unknowns of the model is given by

$$p(\boldsymbol{\theta}, \mathbf{s} \mid \mathbf{y}, \mathbf{X}) \quad (11)$$

The joint posterior is not tractable, we approximate using the full conditionals for each of the parameters.

We propose a MCMC procedure based on Gibbs sampling to sample the unknowns from 3 **blocks**.

- Block 1: Sampling $\beta_{m,j_m}^{(d)}, \gamma_{m,j_m}^{(d)}, \sigma_m^2, \sigma^2, \mu$ from $p(\beta_{m,j_m}^{(d)}, \gamma_{m,j_m}^{(d)}, \sigma_m^2, \sigma^2, \mu \mid \mathbf{Y}, \mathbf{X}_1, \dots, \mathbf{X}_T)$
- Block 2: Sampling $\zeta^{(d)}$ and τ from $p(\zeta^{(d)}, \tau \mid \mathbf{B}, \gamma, \mathbf{w})$
- Block 3: Sampling $\lambda_m^{(d)}$ and $\mathbf{w}_{m,j_m}^{(d)}$ from $p(\lambda_m^{(d)}, \mathbf{w}_{m,j_m}^{(d)} \mid \gamma_{m,j_m}^{(d)}, \tau, \zeta^{(d)})$

For the hidden states, we apply a **Forward Filtering Backward Sampling** (FFBS) strategy:

- Draw transitional probabilities (p_{1k}, \dots, p_{Kk}) from Dirichlet distribution $p(p_{1k}, \dots, p_{Kk} \mid \mathbf{s})$.
- Compute iteratively the vector of smoothed probabilities $\xi_{t|T}$ by using Hamilton Filter, and draw the state vector \mathbf{s}_t from a multinomial distribution $\mathcal{M}(\mathbf{1}, \xi_{t|T})$.

Proposition 1: Backfitting

The model in Eq.(1) can be written as:

$$y_t = \beta_{m,j_m}^{(d)'} \psi_{j_m d t} + R_{j_m d t} + R_{d t} + \sigma^2 \varepsilon_t$$

where the residual terms $R_{d t}$, $R_{j_m d t}$ and auxiliary covariate vector $\psi_{j_m d t}$ are:

$$R_{d t} = \sum_{d' \neq d} \langle B_1^{(d')} \circ \dots \circ B_M^{(d')}, X_t \rangle$$

$$R_{j_m d t} = \langle (B_1^{(d)} \circ \dots \circ B_M^{(d)})_{-j_m}, (X_t)_{-j_m} \rangle$$

$$\psi_{j_m d t} = \text{vec}(B_1^{(d)} \circ \dots \circ B_{m-1}^{(d)} \circ B_{m+1}^{(d)} \circ \dots \circ B_M^{(d)} \circ X_t)_{\tilde{j}_m}$$

MCMC-Random Partial Scan Gibbs

For the first 10 Gibbs iterations, we run full scan for every rank and every mode to recover the main structure of the coefficients. Then we perform Random-Partial-Scan Gibbs to randomly select a subset of components to update for each iteration.

Algorithm The steps in a Random Partial Scan Gibbs

- 1: Draw uniformly $J \subset \{1, \dots, n\}$ a random set of indices of size $n^* \leq n$ so that each subset has an equal chance of being selected.
 - 2: If $J = (j_1, \dots, j_{n^*})$, update $\theta_J = (\theta_{j_1}, \theta_{j_2}, \dots, \theta_{j_{n^*}})$ using a random scan and leave the other components of θ unchanged.
-

► performance

Simulations

Tensor Regression

Simulation settings:

- 4 experimental settings ranging from different ranks and **different levels of sparsity**.
- Matrix predictor with dimensions 20×20
- Number of observations: 400
- Gibbs iterations: 3000

► robustness checks

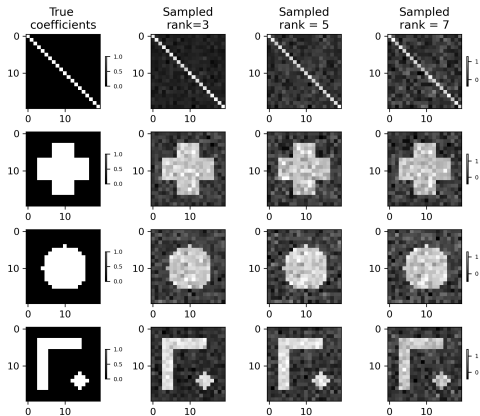


Figure: Estimated coefficients for four experimental settings using three different ranks

Simulations

Tensor Regression

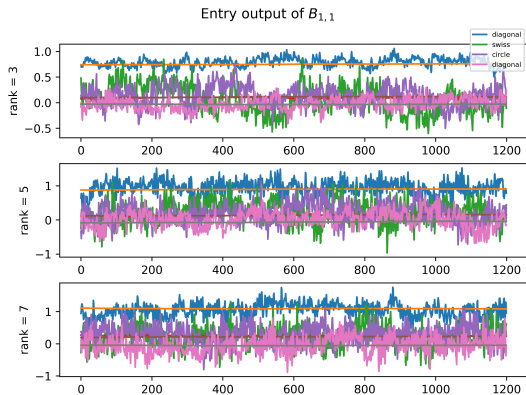


Figure: Raw MCMC output and progressive average of entry $B_{1,1}$ for different types of coefficients

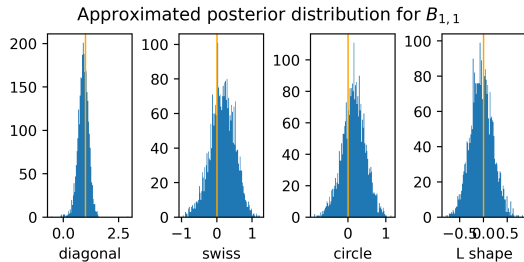


Figure: Approximated posterior distribution

Simulation

Markov Switching

Simulation settings

- 2 sets of true coefficients are used to represent 2 different regimes, both **i.i.d** covariates and **AR(1)** covariates are used in the simulation.
- Matrix predictor with dimensions 12×12
- Regime specific intercepts: $\mu_1 = \mu_2 = 0$
Regime specific variances: $\sigma_1^2 = 2, \sigma_2^2 = 0.1$.
- Number of observations: 800
- Gibbs iterations: 3000

▶ more results

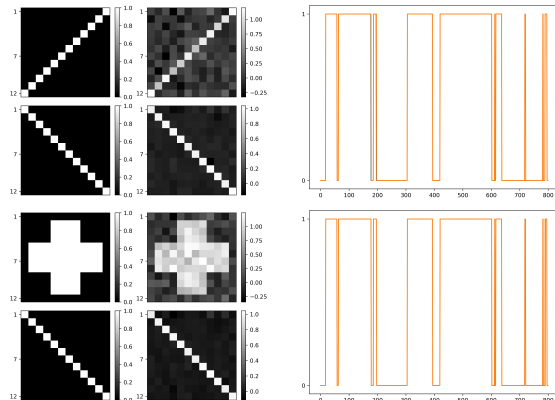


Figure: Markov-switching model with Diagonal and Anti-diagonal coefficients (first row) and with Cross and Diagonal coefficients (second row).

Table: MCMC convergence and efficiency

Setting S_1^{MS} (anti-diag / diag)					
	ACF(1)	ACF(5)	ACF(10)	MSE(10)	MSE(100)
Coefficients (B)	0.4085 (0.3145)	0.3279 (0.2328)	0.3158 (0.0980)	0.0559	0.0083
States(s_t)	0.5624 (0.5448)	0.5437 (0.3878)	0.5333 (0.1942)	0.2725	0.0113
Setting S_2^{MS} (cross / diag)					
Coefficients (B)	0.5139 (0.4247)	0.4425 (0.2819)	0.4410 (0.1650)	0.1773	0.0106
States(s_t)	0.5294 (0.5153)	0.5166 (0.3649)	0.5077 (0.1831)	0.3013	0.0050

Table 1 documents the results on convergence for the two different experimental settings. The second column of the table reports the ACFs of the parameters and the hidden states before and after thinning, where the results after thinning are reported in parentheses. The third column reports the MSE of the parameters and hidden states at the 10th and 100th Gibbs iteration.

Simulation

Model Selection

To understand how the [number of components](#) and [regimes](#) affects the inference, we conducted further simulation experiments with 3 regimes and an increasing number of components $D = 3, 5, 7$.

We evaluate the performance of different models with different combinations of components and regimes using the [Watanabe-Akaike Information Criterion \(WAIC\)](#) (Watanabe and Oppen, 2010) to provide some guidelines on choosing D and K .

	$D = 3$	$D = 5$	$D = 7$
$K = 2$	2211.74	2613.94	2558.18
$K = 3$	4095.90	4169.02	4257.44

Table: WAIC-based model comparison for Markov-Switching Tensor Regression. D is the number of components for tensor decomposition, and K is the number of regimes. The model with the best performance is shown in boldface.

Oil and Stock Daily volatility – Variables

Main variables

Daily volatility index of the US market (VIX) and the crude oil ETF volatility index (OVX). VIX measures the market's expectations and sentiments; thus, predicting VIX is crucial for developing investment strategies.

Heterogenous Autoregressive (HAR)

Fernandes et al. (2014) studied the long-range dependence in the VIX data by including a vector of the average of the logarithm of VIX for the last $h \in \{1, 5, 10, 22, 66\}$ days (to mirror daily, weekly, bi-weekly, monthly and quarterly component) in a family of heterogeneous autoregressive (HAR) processes.

Oil and Stock Daily volatility - Model

Our model

We extend it to a multiple-equation tensor regression framework: VIX on OVX and OVX on VIX. Define $y_{1,t} = \text{VIX}_t$ and $y_{2,t} = \text{OVX}_t$

We include other covariates: the h day log-return for S&P 500 (SP_{t-h}), exchange rate (proxy by US dollar index, ER_{t-h}), spot price of WTI crude oil (Oil_{t-h}) for $h \in \{1, \dots, 44\}$.

Covariate tensor for each response variable is a 4×44 matrix.

$$\begin{cases} \text{VIX}_t = \mu_1(s_t) + \left\langle B_1(s_t), \begin{pmatrix} SP_{t-1} & \dots & SP_{t-h} & \dots & SP_{t-44} \\ ER_{t-1} & \dots & ER_{t-h} & \dots & ER_{t-44} \\ Oil_{t-1} & \dots & Oil_{t-h} & \dots & Oil_{t-44} \\ OVX_{t-1} & \dots & OVX_{t-h} & \dots & OVX_{t-44} \end{pmatrix} \right\rangle + \sigma_1(s_t)\varepsilon_{1t}, \\ \text{OVX}_t = \mu_2(s_t) + \left\langle B_2(s_t), \begin{pmatrix} SP_{t-1} & \dots & SP_{t-h} & \dots & SP_{t-44} \\ ER_{t-1} & \dots & ER_{t-h} & \dots & ER_{t-44} \\ Oil_{t-1} & \dots & Oil_{t-h} & \dots & Oil_{t-44} \\ VIX_{t-1} & \dots & VIX_{t-h} & \dots & VIX_{t-44} \end{pmatrix} \right\rangle + \sigma_2(s_t)\varepsilon_{2t}. \end{cases} \quad (12)$$

Oil and Stock Daily volatility - Predictive ability

Models	WAIC	In-sample		Out-of-sample			
		MSE	MAE	$h = 1$		$h = 5$	
				MSE	MAE	MSE	MAE
$TR(2)$	3231.06	0.3097	0.4324	0.2540	0.4232	0.3581	0.5182
MSTR(2, 2)	1907.28	0.0892	0.2376	0.1409	0.3342	0.1379	0.3063
$MSTR(3, 2)$	911.11	0.0339	0.1447	0.3199	0.4024	0.1976	0.3388
$TR(3)$	3282.48	0.3445	0.4534	0.2172	0.4155	0.2905	0.4751
MSTR(2, 3)	2347.39	0.1019	0.2505	0.0939	0.2641	0.0659	0.2132
MSTR(3, 3)	518.13	0.0272	0.1221	0.9328	0.9398	0.3471	0.5236
LS	—	0.3049	0.4266	0.1945	0.3474	0.3668	0.5211
$LASSO$	—	0.4207	0.5259	0.5199	0.6363	0.6940	0.7589

VIX Equation - MSTR and LASSO

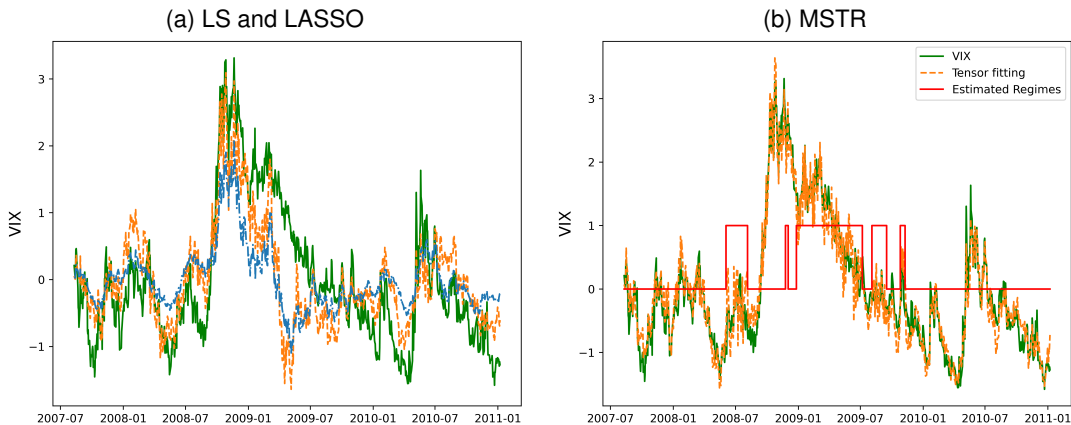


Figure: Left: In-sample fitting for Least Squares (orange dashed) and LASSO (blue dashed). Right: In-sample fitting of the Markov-Switching Tensor Regression model MSTR(2, 2) (orange dashed) and estimated hidden states (red solid). Actual VIX (green solid).

VOX Equation - MSTR and LASSO

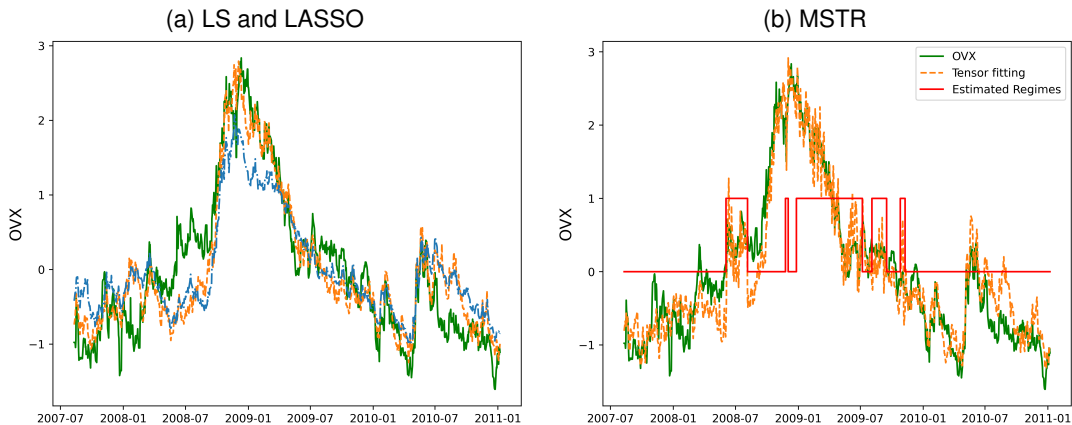


Figure: Left: In-sample fitting for Least Squares (orange dashed) and LASSO (blue dashed). Right: In-sample fitting of the Markov-Switching Tensor Regression model MSTR(2, 2) (orange dashed) and estimated hidden states (red solid). Actual VOX (green solid).

Asymmetric Effects - Impact of Oil

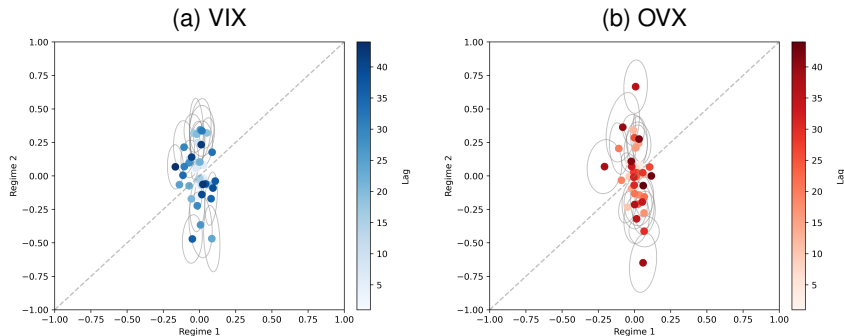


Figure: MSTR(2, 2). Effects of h -day Oil log-returns, $h \in \{1, \dots, 44\}$. Lighter and darker colors: smaller and larger h , respectively. 90% HPD (gray ellipses) only for the coefficients with asymmetric effects (ellipse does not intersect the 45° line).

- (1) Limited impact in the low-volatility regime.
- (2) Stronger effect on OVX than on VIX in the high-volatility regime.
- (3) Long-term effects (dark red) on VOX (Bandi and Perron, 2006; Corsi, 2009).

Asymmetric Effects - Impact of S&P 500

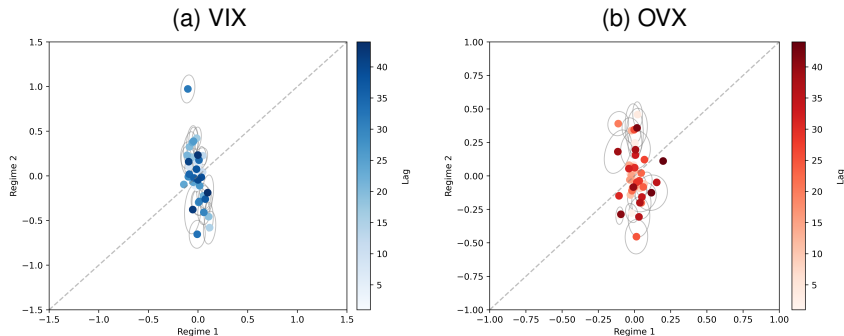


Figure: MSTR(2, 2). Effects of h -day S&P 500 log-returns $h \in \{1, \dots, 44\}$. Lighter and darker colors: smaller and larger h , respectively. 90% HPD (gray ellipses) only for the coefficients with asymmetric effects (ellipse does not intersect the 45° line).

- (1) Limited effect in the low-volatility regime.
- (2) Medium lags have a larger effect than lower and higher lags.
- (3) Larger impact on VIX.

Concluding Remarks

- A new Markov switching multiple-equation tensor regression model capable of extracting a common latent factor (latent regime changes) is proposed.
- A low-rank representation of the coefficient tensor and hierarchical prior distribution are proposed to introduce shrinkage effects to overcome overparametrization.
- An efficient MCMC sampler is proposed based on back-fitting and random scan strategies.
- The tensor regression model is readily to be used with tensor covariates with order 2 or 3.

Concluding Remarks

- A new Markov switching multiple-equation tensor regression model capable of extracting a common latent factor (latent regime changes) is proposed.
- A low-rank representation of the coefficient tensor and hierarchical prior distribution are proposed to introduce shrinkage effects to overcome overparametrization.
- An efficient MCMC sampler is proposed based on back-fitting and random scan strategies.
- The tensor regression model is readily to be used with tensor covariates with order 2 or 3.

Thanks!!!

- Bandi, F. M. and Perron, B. (2006). Long memory and the relation between implied and realized volatility. *Journal of Financial Econometrics*, 4(4):636–670.
- Billio, M., Casarin, R., and Iacopini, M. (2024). Bayesian Markov-switching tensor regression for time-varying networks. *Journal of the American Statistical Association*, 119(545):109–121.
- Billio, M., Casarin, R., Iacopini, M., and Kaufmann, S. (2023). Bayesian dynamic tensor regression. *Journal of Business & Economic Statistics*, 41(2):429–439.
- Corsi, F. (2009). A simple approximate long-memory model of realized volatility. *Journal of Financial Econometrics*, 7(2):174–196.
- Fernandes, M., Medeiros, M. C., and Scharth, M. (2014). Modeling and predicting the cboe market volatility index. *Journal of Banking & Finance*, 40:1–10.

References II

- Guha, S. and Rodriguez, A. (2021). Bayesian regression with undirected network predictors with an application to brain connectome data. *Journal of the American Statistical Association*, 116(534):581–593.
- Kolda, T. G. and Bader, B. W. (2009). Tensor decompositions and applications. *SIAM Review*, 51(3):455–500.
- Papadogeorgou, G., Zhang, Z., and Dunson, D. B. (2021). Soft tensor regression. *Journal of Machine Learning Research*, 22:219–1.
- Spencer, D., Guhaniyogi, R., Shinohara, R., and Prado, R. (2022). Bayesian tensor regression using the Tucker decomposition for sparse spatial modeling. *arXiv preprint arXiv:2203.04733*.
- Wang, K. and Xu, Y. (2024). Bayesian tensor-on-tensor regression with efficient computation. *Statistics and its Interface*, 17(2):199.

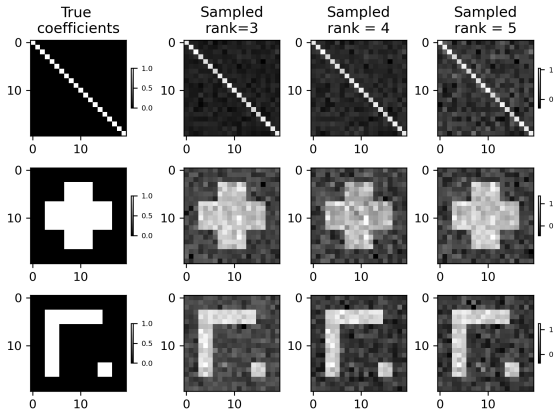
Watanabe, S. and Opper, M. (2010). Asymptotic equivalence of bayes cross validation and widely applicable information criterion in singular learning theory. *The Journal of Machine Learning Research*, 11(12).

Robustness Check

We tweaked a bit with hyperparameters to change the prior mean and variance of the scales while still maintaining $V^* = 1, AV^* = 10\%$.

	benchmark	robustness
α	1	1
a_σ	0.5	0.5
b_σ	$8.5\sqrt{C}$	$2\sqrt{C}$
a_τ	3	3
b_τ	$33.75\sqrt{C}/b_\sigma$	$33.75\sqrt{C}/b_\sigma$
a_λ	3	3
b_λ	$a_\lambda^{1/4}$	$a_\lambda^{1/2}$

◀ back



Robustness Check

Noisy True Coefficients

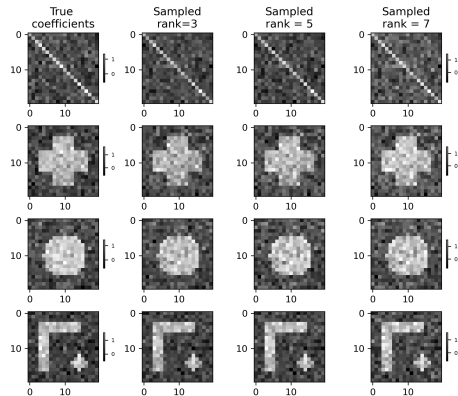


Figure: Estimation results with noisy true coefficients

Simulations Results

Computational cost

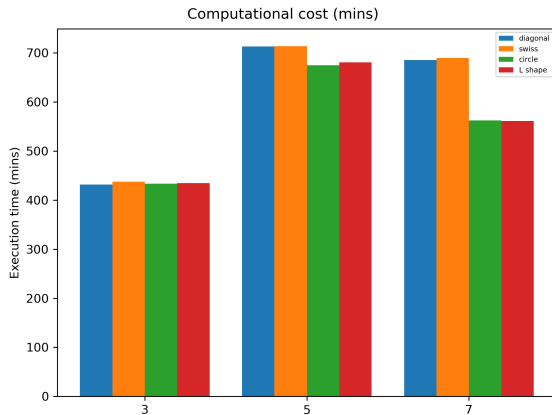


Figure: Computational cost (mins)

Simulations Results

Autocorrelation

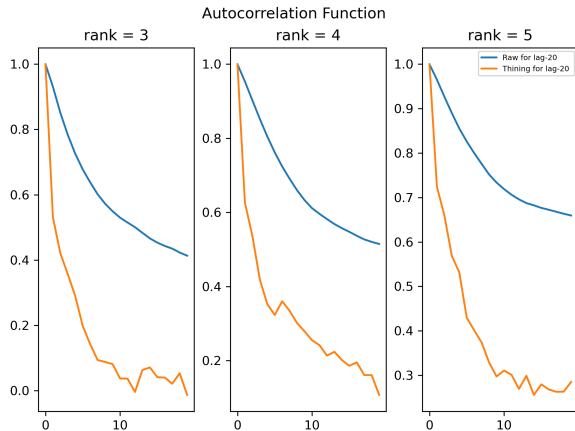


Figure: Autocorrelation before and after thining

Simulation Results

Intercepts and Variance of MS

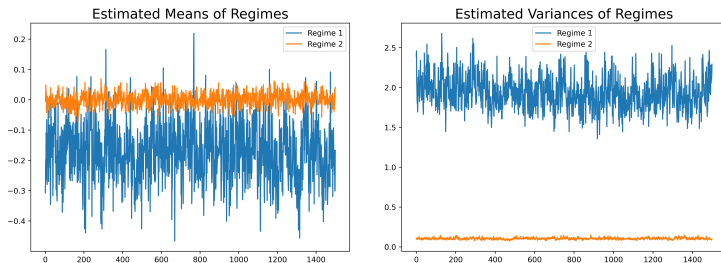


Figure: Trace plots after removing the burn-in samples for regime specific intercepts (left) and variances (right) for the S_2^{MS} experimental setting. True values are $\mu_1 = \mu_2 = 0$, $\sigma_1^2 = 2$, $\sigma_2^2 = 0.1$.

◀ back

Random Partial Scan Gibbs

◀ back

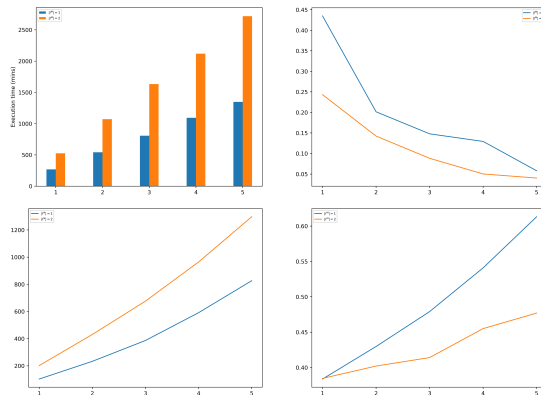


Figure: Upper left: computational cost (CPU time in minutes) for $|I^D| = 1, \dots, 5$ (horizontal axis) and $|I^M| = 1$ (blue bar), $|I^M| = 2$ (orange bar). Upper right: Autocorrelation at the 10-th lag for different $(|I^D|, |I^M|)$. Bottom left: Effective sample size. Bottom right: Effective sample size per computational unit (minute).