Sparse covariance models for high dimensional image data analysis

Alfred Hero

Dept of Electrical Engineering and Computer Science (EECS), Dept of Biomedical Engineering (BME), Dept of Statistics
Program in Applied and Interdisciplinary Mathematics
Program in Applied Physics
Program in Computational Medicine and Bioinformatics
University of Michigan - Ann Arbor

Dec. 4 , 2020

References:

Principal references for theory presented here

1. Kristjian Greenewald and Alfred Hero, *Robust Kronecker Product PCA for Spatio-Temporal Covariance Estimation*, IEEE Transactions on Signal Processing, vol 63, no 23, pp. 6368-6378, Dec. 2015. arxiv:1411.1352.
2. Kristjian Greenewald, Shuheng Zhou, Alfred Hero (2019), *The Tensor Graphical Lasso (TeraLasso)*, Journal of the Royal Statistical Society, Series B, 81:5, pp. 901-931, November 2019. arxiv:1705.03983.
3. Yu Wang, Byoung Jang, Alfred Hero, *The Sylvester Graphical Lasso (SyGlasso)*, AISTATS June 2020. arxiv:2002.00288
4. Yu Wang and Alfred Hero, *A proximal alternating linearized minimization method for tensor graphical models*, submitted.

References:

Principal references for theory presented here

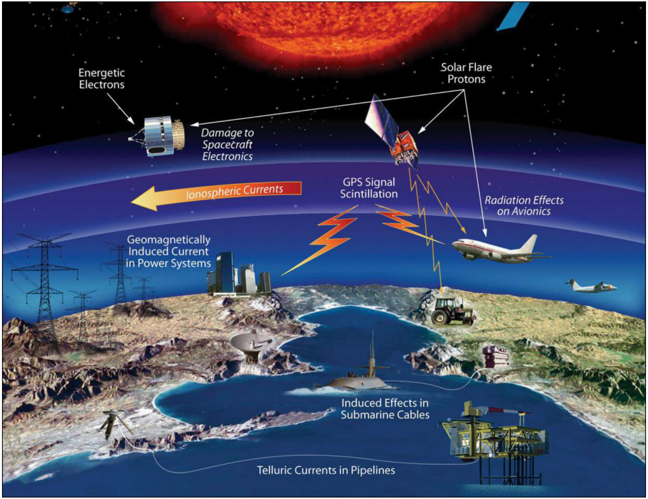
1. Kristjian Greenewald and Alfred Hero, *Robust Kronecker Product PCA for Spatio-Temporal Covariance Estimation*, IEEE Transactions on Signal Processing, vol 63, no 23, pp. 6368-6378, Dec. 2015. arxiv:1411.1352.
2. Kristjian Greenewald, Shuheng Zhou, Alfred Hero (2019), *The Tensor Graphical Lasso (TeraLasso)*, Journal of the Royal Statistical Society, Series B, 81:5, pp. 901-931, November 2019. arxiv:1705.03983.
3. Yu Wang, Byoung Jang, Alfred Hero, *The Sylvester Graphical Lasso (SyGlasso)*, AISTATS June 2020. arxiv:2002.00288
4. Yu Wang and Alfred Hero, *A proximal alternating linearized minimization method for tensor graphical models*, submitted.

Research supported by

1. NASA/NSF: Solar Storms and Terrestrial Impacts Center (SOLSTICE)
2. ARO MURI: Non-commutative information
3. AFOSR: ATR-Center Program
4. DOE: Consortium for Exploitation of Technological Innovations
5. DARPA: Guaranteeing AI Robustness Against Deception (GARD)

Space Weather can have significant terrestrial impact[1]

Figure 1. Examples of Potential Effects of Space Weather



Source: NASA, email communication with NASA Office of Legislative and Intergovernmental Affairs, September 6, 2019.

[1] *Space Weather: An Overview of Policy and Select U.S. Government Roles and Responsibilities*, Congressional Research Service Report, Jan. 2020

Space Weather can have significant terrestrial impact

In 2019 FEMA National Threat and Hazard Identification and Risk Assessment (THIRA) Report[2] cites Space Weather among top 5 threats
Space Weather places 3rd among threats with nationwide impact

Table 1: Threats and Hazards of Concern Identified for the *2019 National THIRA*[8,9]

Threat/Hazard Type	Threat/Hazard	Area/Region
Natural	Plausible Concurrent Operations[10]	Nationwide
	Earthquake	Washington, Oregon, California, Idaho
		600,000 sq. km in the Midwest/East
	Hurricane	Galveston, Texas to the Midwest
		Fort Lauderdale, Florida to Alabama
		Hawaii
	Pandemic	Nationwide
	Space Weather	Nationwide

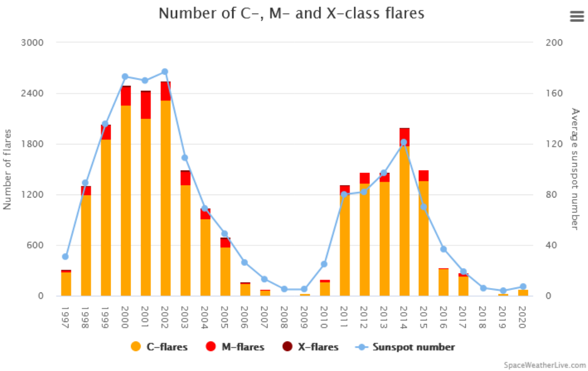
[10] Plausible concurrent operations: response capacity overwhelmed in Aug and Sept 2019 when FEMA responded to 3 hurricanes (Harvey-Texas, Irma-Florida and Maria-Puerto-Rico), 5 flooding events across nation, and 1 major CA wildfire.

[2] National Threat and Hazard Identification and Risk Assessment (THIRA), *FEMA DHS Report, July 2019*

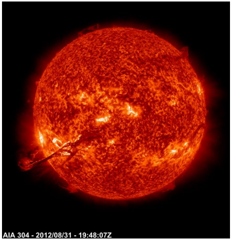
Solar imaging - active regions, coronal mass ejections, and flares

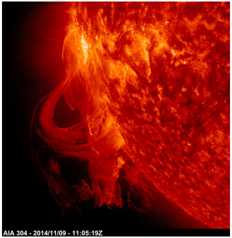
Flares and coronal mass ejections (CME) are generated from active regions (sunspots)

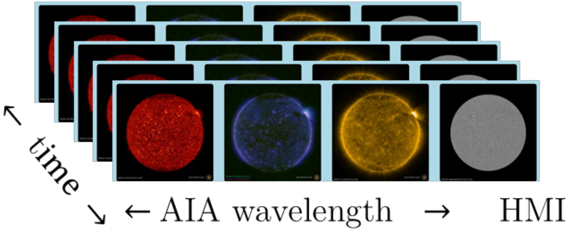
- Active regions evolve over time, space and channel
- Some active regions produce flares, some active regions are quiet.
- Solar flares are classified into different intensity categories: A, B, C, M, X



Predicting and identifying solar CMEs from satellite image data[3] [4]







[3]Y. Chen, W. Manchester, A.O. Hero . . . , (2019), *Identifying solar flare precursors using time series of SDO/HMI images and SHARP parameters*, Space Weather, Oct 2019. .

[4]Z. Jiao, H Sun, X Wang, W Manchester, AO Hero, and Y. Chen, *Solar Flare Intensity Prediction with Machine Learning Models*, Space Weather, May 2020.

Learning multiway data representations

Objective: Learn a low dimensional representation of a multi-modal data cube

$$\mathbf{Z} = \{Z_{i_1,\ldots,i_K}\}_{i_1,\ldots,i_K}^{d_1,\ldots,d_K}, \qquad d = \prod_{i=1}^{K} d_i, \qquad m_k = d/d_k = \prod_{i \neq k}^{K} d_k$$

[5] Kolda and Bader, *Tensor decompositions and applications*, SIAM Review, 51(3):445-500, 2009

Learning multiway data representations

Objective: Learn a low dimensional representation of a multi-modal data cube

$$\mathbf{Z} = \{Z_{i_1,\ldots,i_K}\}_{i_1,\ldots,i_K}^{d_1,\ldots,d_K}, \qquad d = \prod_{i=1}^{K} d_i, \qquad m_k = d/d_k = \prod_{i \neq k}^{K} d_k$$

Common traits of data cube (K-way tensor):

- the indices may index over different mode types.
- some mode dimensions may be high, others low.
- the data cube may be sparse in some modes.
- data cube may have sparse, low dimensional approximation

[5] Kolda and Bader, *Tensor decompositions and applications*, SIAM Review, 51(3):445-500, 2009

Learning multiway data representations

Objective: Learn a low dimensional representation of a multi-modal data cube

$$\mathbf{Z} = \{Z_{i_1,\ldots,i_K}\}_{i_1,\ldots,i_K}^{d_1,\ldots,d_K}, \qquad d = \prod_{i=1}^{K} d_i, \qquad m_k = d/d_k = \prod_{i \neq k}^{K} d_k$$

Common traits of data cube (K-way tensor):

- the indices may index over different mode types.
- some mode dimensions may be high, others low.
- the data cube may be sparse in some modes.
- data cube may have sparse, low dimensional approximation

First order representations: learning the mean[5].

$$\mathbb{E}[\mathbf{Z}] \approx \mathbf{a} \circ \mathbf{b} \circ \mathbf{c}, \quad \mathbb{E}[\mathbf{Z}] \approx \mathbf{A} \otimes \mathbf{B} \otimes \mathbf{C}, \quad \mathbb{E}[\mathbf{Z}] \approx \mathbf{A} \oplus \mathbf{B} \oplus \mathbf{C},$$

[5] Kolda and Bader, *Tensor decompositions and applications*, SIAM Review, 51(3):445-500, 2009

Learning multiway data representations

Objective: Learn a low dimensional representation of a multi-modal data cube

$$\mathbf{Z} = \{Z_{i_1,\ldots,i_K}\}_{i_1,\ldots,i_K}^{d_1,\ldots,d_K}, \qquad d = \prod_{i=1}^{K} d_i, \qquad m_k = d/d_k = \prod_{i \neq k}^{K} d_k$$

Common traits of data cube (K-way tensor):

- the indices may index over different mode types.
- some mode dimensions may be high, others low.
- the data cube may be sparse in some modes.
- data cube may have sparse, low dimensional approximation

First order representations: learning the mean[5].

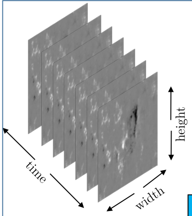
$$\mathbb{E}[\mathbf{Z}] \approx \mathbf{a} \circ \mathbf{b} \circ \mathbf{c}, \quad \mathbb{E}[\mathbf{Z}] \approx \mathbf{A} \otimes \mathbf{B} \otimes \mathbf{C}, \quad \mathbb{E}[\mathbf{Z}] \approx \mathbf{A} \oplus \mathbf{B} \oplus \mathbf{C},$$

Second order representations: learning the covariance or inverse covariance

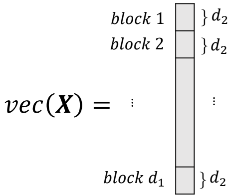
$$\mathrm{cov}(\mathbf{Z}) \approx \mathbf{A} \otimes \mathbf{B}, \quad \mathrm{cov}^{-1}(\mathbf{Z}) \approx \mathbf{A} \oplus \mathbf{B}$$

[5] Kolda and Bader, *Tensor decompositions and applications*, SIAM Review, 51(3):445-500, 2009

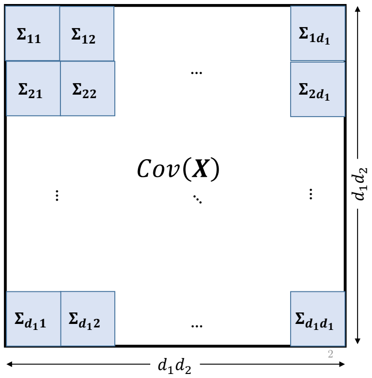
Multiway data structure = patterned covariance



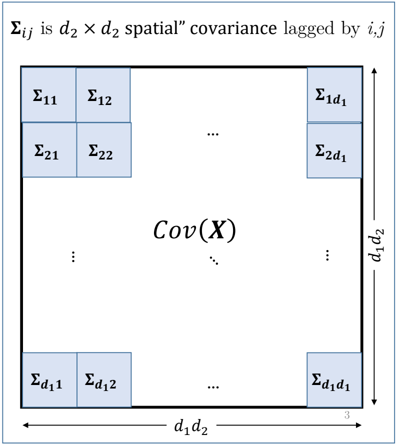
$$\boldsymbol{X} = \begin{matrix} X_{11} & \cdots & X_{1d_2} \\ \vdots & \cdots & \vdots \\ X_{1d_2} & \cdots & X_{d_1 d_2} \end{matrix}$$

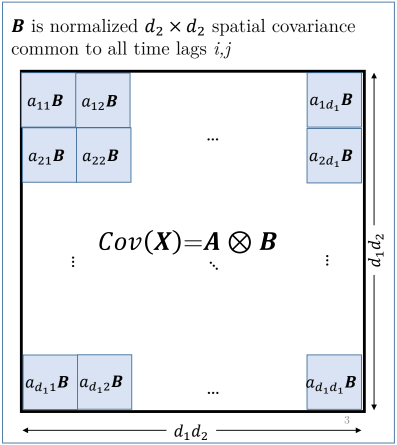


$\boldsymbol{\Sigma}_{ij}$ is $d_2 \times d_2$ spatial" covariance lagged by i,j



Multiway data structure ≈ Kronecker product covariance





Processes obeying diffusion equations have sparse KP inverse covariance

Consider a random process $u(t, x, y)$ whose evolution is governed by a pde, e.g.,

$$\frac{du}{dt} = a\partial^2_{xx} u + b\partial^2_{yy} u + c\partial^2_{xy} u + w.$$

$w = w(t, x, y)$ is a random disturbance.

Processes obeying diffusion equations have sparse KP inverse covariance

Consider a random process $u(t, x, y)$ whose evolution is governed by a pde, e.g.,

$$\frac{du}{dt} = a\partial_{xx}^2 u + b\partial_{yy}^2 u + c\partial_{xy}^2 u + w.$$

$w = w(t, x, y)$ is a random disturbance.

Differential-operator equivalent equation

$$(D_t \otimes I \otimes I)\mathbf{u} = (a[I \otimes D_{xx} \otimes I] + b[I \otimes I \otimes D_{yy}] + c[I \otimes D_x \otimes D_y])\mathbf{u} + \mathbf{w}$$

Euler discretized form described by KP time-space matrix equation:

$$(\mathbf{D}_t \otimes \mathbf{D}_z)\mathbf{u} = \mathbf{w} \quad \Leftrightarrow \quad \mathbf{D}_t\mathbf{U}\mathbf{D}_z = \mathbf{W}$$

- $\mathbf{U} = \{u(t, \mathbf{z})\}_{t\in[0,T],\mathbf{z}\in\mathbb{R}^2}$, $\mathbf{u} = \text{vec}(\mathbf{U})$
- $\mathbf{D}_t$ and $\mathbf{D}_z$ are invertible sparse tridiagonal and pentadiagonal matrices.
- For $\mathbf{w}$ white noise, $\text{var}(\mathbf{w}) = \sigma^2$, the inverse covariance of $\mathbf{U}$ is sparse KP

$$\mathbf{\Sigma}^{-1} = \sigma^{-2}\mathbf{D}_t \otimes \mathbf{D}_z$$

Processes obeying diffusion equations have sparse KP inverse covariance

Consider a random process $u(t, x, y)$ whose evolution is governed by a pde, e.g.,

$$\frac{du}{dt} = a\partial_{xx}^2 u + b\partial_{yy}^2 u + c\partial_{xy}^2 u + w.$$

$w = w(t, x, y)$ is a random disturbance.

Differential-operator equivalent equation

$$(D_t \otimes I \otimes I)\mathbf{u} = (a[I \otimes D_{xx} \otimes I] + b[I \otimes I \otimes D_{yy}] + c[I \otimes D_x \otimes D_y])\mathbf{u} + \mathbf{w}$$

Euler discretized form described by KP time-space matrix equation:

$$(\mathbf{D}_t \otimes \mathbf{D}_z)\mathbf{u} = \mathbf{w} \quad \Leftrightarrow \quad \mathbf{D}_t \mathbf{U} \mathbf{D}_z = \mathbf{W}$$

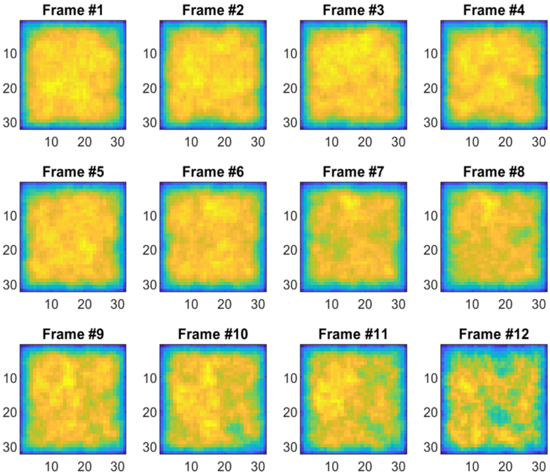
- $\mathbf{U} = \{u(t, \mathbf{z})\}_{t \in [0,T], \mathbf{z} \in \mathbb{R}^2}$, $\mathbf{u} = \text{vec}(\mathbf{U})$
- $\mathbf{D}_t$ and $\mathbf{D}_z$ are invertible sparse tridiagonal and pentadiagonal matrices.
- For $\mathbf{w}$ white noise, $\text{var}(\mathbf{w}) = \sigma^2$, the inverse covariance of $\mathbf{U}$ is sparse KP

$$\mathbf{\Sigma}^{-1} = \sigma^{-2}\mathbf{D}_t \otimes \mathbf{D}_z$$

Example: diffusive limit of the dissipative magnetic induction equation

$$\frac{du}{dt} - \eta\nabla^2 u = -\alpha u + w$$

Generated sample from dissipative magnetic induction equation



Simulation of a single realization of $\mathbf{U}$

$(\eta = 1, \alpha = 0.6)$

Kronecker product expansions for covariance matrices

Any symmetric $d_1 d_2 \times d_1 d_2$ matrix $\mathbf{\Lambda}$ has the KPCA representation[6] [7]

$$\mathbf{\Lambda} = \sum_{i=1}^{r} \sigma_i \mathbf{A}_i \otimes \mathbf{B}_i$$

$w = w(t, x, y)$ is a random disturbance. for some sequence $\{\mathbf{A}_i, \mathbf{B}_i\}_{i=1}^{r}$ of $d_1 \times d_1$ and $d_2 \times d_2$ matrices

- $r \leq \min\{d_1, d_2\}$ is *separation rank* of $\mathbf{\Lambda}$
- $\sigma_i > 0$ are (KPCA) coefficients
- $\|\mathbf{A}_i\|_F = \|\mathbf{B}_i\|_F = 1$

Compare to PCA representation

$$\mathbf{\Lambda} = \sum_{i=1}^{d_1 d_2} \lambda_i \mathbf{u}_i \mathbf{u}_i^T$$

[6] Tsiligkaridis and Hero, *Covariance Estimation in High Dimensions via Kronecker Product Expansions*, IEEE Trans on Signal Processing, Vol 61, No. 21, pp. 5347 5360, Nov 2013. arxiv 1302.2686.

[7] Greenewald and Hero, *Robust Kronecker Product PCA for Spatio-Temporal Covariance Estimation*, IEEE Transactions on Signal Processing, vol 63, no 23, pp. 6368-6378, Dec. 2015. arxiv:1411.1352.

KPCA vs PCA for dissipative magnetic induction process

$d_2 = 32 \times 32$ image pixels, $d_1 = 12$ time samples, $n = 1000$ samples

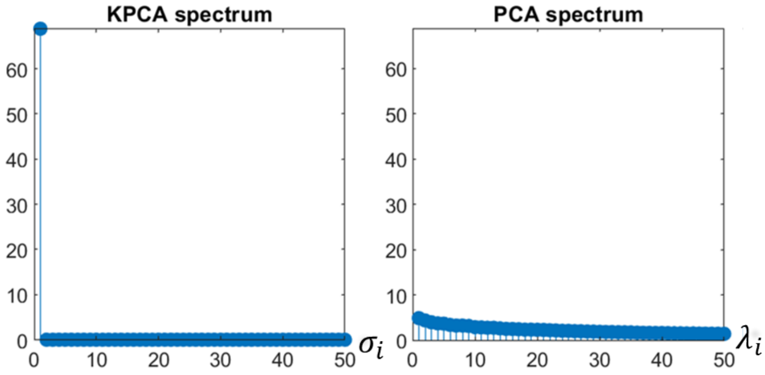


Figure: *KPCA vs PCA spectrum of population covariance* $\boldsymbol{\Sigma}$

- Principal KPCA component fits 100% of $\boldsymbol{\Sigma}$
- Principal PCA component fits 2% of $\boldsymbol{\Sigma}$

KPCA vs PCA for dissipative magnetic induction process

$d_2 = 32 \times 32$ image pixels, $d_1 = 12$ time samples, $n = 1000$ samples

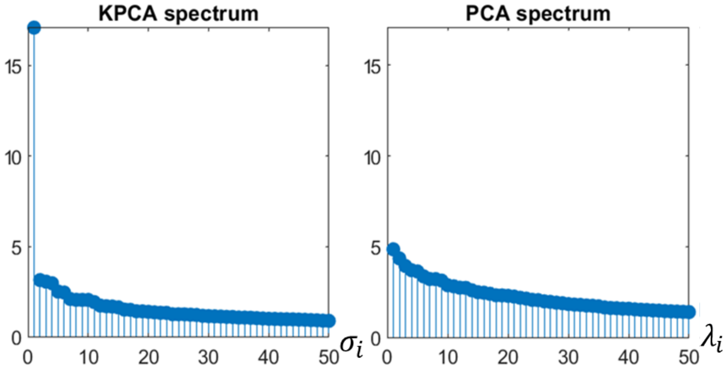


Figure: *KPCA vs PCA spectrum of sample covariance* $\mathbf{S}_n$

- Principal KPCA component fits 82% of $\mathbf{S}_n$
- Principal PCA component fits 1% of $\mathbf{S}_n$

KPCA vs PCA for dissipative magnetic induction process

$d_2 = 32 \times 32$ image pixels, $d_1 = 12$ time samples, $n = 1000$ samples

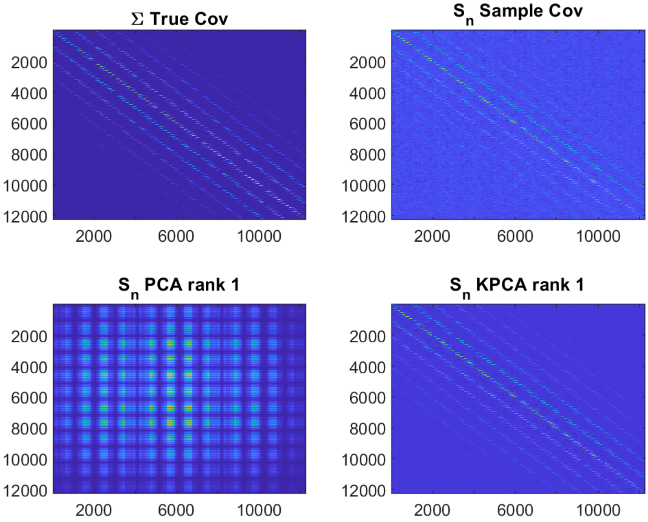


Figure: *KPCA and PCA principal components of sample covariance* $\mathbf{S}_n$

KPCA factors for dissipative magnetic induction process

$d_2 = 32 \times 32$ image pixels, $d_1 = 12$ time samples, $n = 1000$ samples

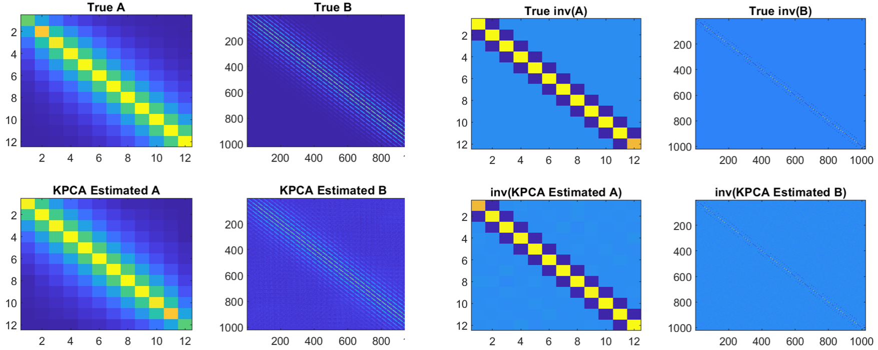


Figure: *Principal KPCA factors for* $\mathbf{S}_n$ *and inv*($\mathbf{S}_n$)

KPCA factors for dissipative magnetic induction process

$d_2 = 32 \times 32$ image pixels, $d_1 = 12$ time samples, $n = 1000$ samples

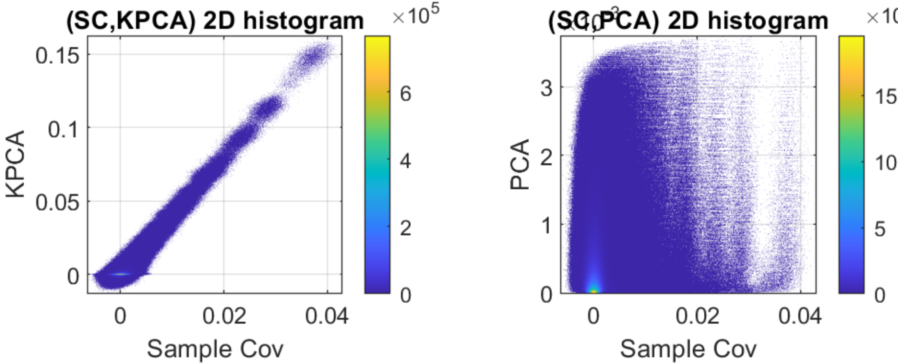


Figure: *Scatterplots of entries* $(\mathbf{S}_n)_{ij}$ *vs entries of rank 1 KPCA* $\sigma_1(\mathbf{A}_1 \otimes \mathbf{B}_1)_{ij}$ *(left) and entries of rank 1 PCA* $\lambda_1(\mathbf{u}_1\mathbf{u}_2^T)_{ij}$ *(right)*

Learning inverse covariance

Many processes do not have sparse covariance $\mathbf{\Sigma}$ but do have sparse *inverse covariance* $\mathbf{\Omega}$

$\Rightarrow$ It is more natural to directly estimate inverse covariance $\mathbf{\Omega} = \mathbf{\Sigma}^{-1}$

Learning inverse covariance

Many processes do not have sparse covariance $\mathbf{\Sigma}$ but do have sparse *inverse covariance* $\mathbf{\Omega}$
$\Rightarrow$ It is more natural to directly estimate inverse covariance $\mathbf{\Omega} = \mathbf{\Sigma}^{-1}$

Penalized maximum likelihood estimator of $\mathbf{\Omega}$ under matrix normal model

Let $p : \mathbb{R}^{d \times d} \to \mathbb{R}$. The PML estimator of $\mathbf{\Omega}$ is

$$\hat{\mathbf{\Omega}} = \mathrm{argmin}_{\mathbf{\Omega}} \mathrm{tr}(\mathbf{S}_n \mathbf{\Omega}) - c_n \mathrm{logdet}((\mathbf{\Omega})) + \lambda p(\mathbf{\Omega}) \tag{1}$$

Learning inverse covariance

Many processes do not have sparse covariance $\boldsymbol{\Sigma}$ but do have sparse *inverse covariance* $\boldsymbol{\Omega}$
$\Rightarrow$ It is more natural to directly estimate inverse covariance $\boldsymbol{\Omega} = \boldsymbol{\Sigma}^{-1}$

Penalized maximum likelihood estimator of $\boldsymbol{\Omega}$ under matrix normal model

Let $p : \mathbb{R}^{d\times d} \to \mathbb{R}$. The PML estimator of $\boldsymbol{\Omega}$ is

$$\hat{\boldsymbol{\Omega}} = \mathrm{argmin}_{\boldsymbol{\Omega}} \mathrm{tr}(\mathbf{S}_n \boldsymbol{\Omega}) - c_n \mathrm{logdet}((\boldsymbol{\Omega})) + \lambda p(\boldsymbol{\Omega}) \tag{1}$$

- Sparsity penalties $p(\boldsymbol{\Omega})$ induce Gaussian graphical model structure
  - Sparse cov estimation: Glasso, Friedman *et al* (2009)
  - Sparse Kronecker product (KGlasso): Allen and Tibshirani (2010)
  - Convex correlation selection method (CONCORD): Khare *et al* (2013)
  - Sparse Kronecker sum: Bigraphical lasso (BiGlasso), Zhou *et al* (2014),
  - Scalable BiGlasso: Tensor Glasso (TeraLasso): Greenwald *et al* (2019)
  - Generative Kronecker sum: Sylvester Glasso (SyGlasso), Wang *et al* (2020)

Learning inverse covariance

Many processes do not have sparse covariance $\boldsymbol{\Sigma}$ but do have sparse *inverse covariance* $\boldsymbol{\Omega}$
$\Rightarrow$ It is more natural to directly estimate inverse covariance $\boldsymbol{\Omega} = \boldsymbol{\Sigma}^{-1}$

Penalized maximum likelihood estimator of $\boldsymbol{\Omega}$ under matrix normal model

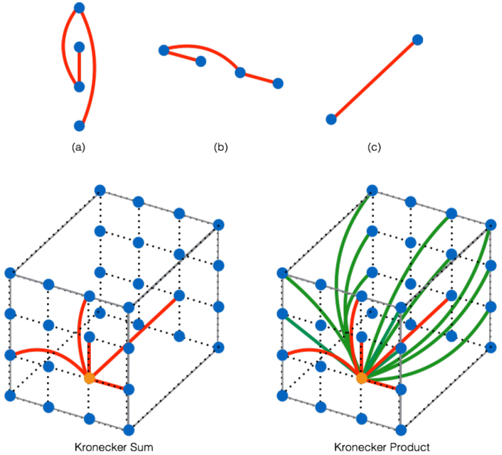
Let $p : \mathbb{R}^{d\times d} \to \mathbb{R}$. The PML estimator of $\boldsymbol{\Omega}$ is

$$\hat{\boldsymbol{\Omega}} = \mathrm{argmin}_{\boldsymbol{\Omega}} \mathrm{tr}(\mathbf{S}_n\boldsymbol{\Omega}) - c_n \mathrm{logdet}((\boldsymbol{\Omega})) + \lambda p(\boldsymbol{\Omega}) \tag{1}$$

- Sparsity penalties $p(\boldsymbol{\Omega})$ induce Gaussian graphical model structure
  - Sparse cov estimation: Glasso, Friedman *et al* (2009)
  - Sparse Kronecker product (KGlasso): Allen and Tibshirani (2010)
  - Convex correlation selection method (CONCORD): Khare *et al* (2013)
  - Sparse Kronecker sum: Bigraphical lasso (BiGlasso), Zhou *et al* (2014),
  - Scalable BiGlasso: Tensor Glasso (TeraLasso): Greenwald *et al* (2019)
  - Generative Kronecker sum: Sylvester Glasso (SyGlasso), Wang *et al* (2020)
- Sparsity models for estimation of $\boldsymbol{\Omega}$ in a nutshell:
  - KGlasso: find sparse $\mathbf{A}, \mathbf{B}$ in model $\boldsymbol{\Omega} = \mathbf{A} \otimes \mathbf{B}$
  - TeraLasso: find sparse $\mathbf{A}, \mathbf{B}$ in model $\boldsymbol{\Omega} = \mathbf{A} \oplus \mathbf{B} = \mathbf{A} \otimes \mathbf{I} + \mathbf{I} \otimes \mathbf{B}$
  - SyGlasso: find sparse model $\mathbf{A}, \mathbf{B}$ in model $\boldsymbol{\Omega} = (\mathbf{A} \oplus \mathbf{B})^2$

Kronecker sum vs Kronecker product

The Kronecker sum model of TeraLasso is sparser than the Kronecker product of KGLasso



$\mathbf{A} \oplus \mathbf{B} \oplus \mathbf{C}$ $\mathbf{A} \otimes \mathbf{B} \otimes \mathbf{C}$

64 total edges 184 total edges

TeraLasso: runtime comparisons for random Erdös-Renyi

K	p	d_k	n	TeraLasso Runtime (s)	BiGLasso Runtime (s)
2	100	10	10	.0131	.84
2	625	25	10	.0147	6.81
2	2500	50	10	.0272	161
2	5625	75	10	.0401	1690
2	10^4	100	10	.0664	
2	2.5×10^5	500	10	1.62	
2	10^6	1000	10	23.2	
2	4×10^6	2000	10	427	
3	10^6	100	10	3.52	NA
3	8×10^6	200	10	11.2	NA
3	1.25×10^8	500	10	32.6	NA
3	1×10^9	1000	10	70.0	NA
4	10^4	10	10	.281	NA
4	1.6×10^5	20	10	.649	NA
4	6.25×10^6	50	10	10.8	NA
4	1.00×10^9	178	10	88.4	NA
5	1.16×10^9	65	10	124	NA

- TeraLasso speedup wrt BiGlasso by 2 to 4 orders of magnitude ($10^2 - 10^4$)
- Teralasso is scalable to many many variables

$$\text{Terralasso/iter} \Rightarrow O(\sum_{k=1}^{K} d_k^3) \qquad \text{Glasso/iter}^{a} \Rightarrow O(p^3) = O\left(\prod_{k=1}^{K} d_k^3\right)$$

[a] Guillot, Rajaratnam, Rolfs, Maleki, Wong (2012). Iterative thresholding algorithm for sparse inverse covariance estimation. *NIPS*.

The Sylvester Glasso

- The model $\mathbf{\Omega} = \mathbf{A} \otimes \mathbf{B}$ is generative and interpretable (diffusion process)
- The model $\mathbf{\Omega} = \mathbf{A} \oplus \mathbf{B} = \mathbf{A} \otimes \mathbf{I} + \mathbf{I} \otimes \mathbf{B}$ is not generative or interpretable
- SyGlasso adopts a generative and interpretable model $\mathbf{\Omega} = (\mathbf{A} \oplus \mathbf{B})^2$

[8] Kressner, Tobler, *Krylov Subspace Methods for Linear Systems with Tensor Product Structure*. SIAM Journal on Matrix Analysis and Application, 2010

The Sylvester Glasso

- The model $\mathbf{\Omega} = \mathbf{A} \otimes \mathbf{B}$ is generative and interpretable (diffusion process)
- The model $\mathbf{\Omega} = \mathbf{A} \oplus \mathbf{B} = \mathbf{A} \otimes \mathbf{I} + \mathbf{I} \otimes \mathbf{B}$ is not generative or interpretable
- SyGlasso adopts a generative and interpretable model $\mathbf{\Omega} = (\mathbf{A} \oplus \mathbf{B})^2$

2-way SyGlasso representation

A data matrix $\mathbf{Z} \in \mathbb{R}^{d_1 \times d_2}$ has a 2-way SyGlasso representation when

1. The inverse covariance matrix $\mathbf{\Omega} = \mathbf{\Sigma}^{-1}$ of $\mathbf{z} =$vec$(\mathbf{Z})$ has the form

$$\mathbf{\Omega} = (\mathbf{A} \oplus \mathbf{B})^2$$

2. The matrix $\mathbf{Z}$ has the stochastic representation, for $\mathbf{W} \sim \mathcal{N}(0, \mathbf{I})$,

$$\mathbf{AZ} + \mathbf{ZB} = \mathbf{W}$$

[8] Kressner, Tobler, *Krylov Subspace Methods for Linear Systems with Tensor Product Structure*. SIAM Journal on Matrix Analysis and Application, 2010

The Sylvester Glasso

- The model $\mathbf{\Omega} = \mathbf{A} \otimes \mathbf{B}$ is generative and interpretable (diffusion process)
- The model $\mathbf{\Omega} = \mathbf{A} \oplus \mathbf{B} = \mathbf{A} \otimes \mathbf{I} + \mathbf{I} \otimes \mathbf{B}$ is not generative or interpretable
- SyGlasso adopts a generative and interpretable model $\mathbf{\Omega} = (\mathbf{A} \oplus \mathbf{B})^2$

2-way SyGlasso representation

A data matrix $\mathbf{Z} \in \mathbb{R}^{d_1 \times d_2}$ has a 2-way SyGlasso representation when

1. The inverse covariance matrix $\mathbf{\Omega} = \mathbf{\Sigma}^{-1}$ of $\mathbf{z} =$vec$(\mathbf{Z})$ has the form

$$\mathbf{\Omega} = (\mathbf{A} \oplus \mathbf{B})^2$$

2. The matrix $\mathbf{Z}$ has the stochastic representation, for $\mathbf{W} \sim \mathcal{N}(0, \mathbf{I})$,

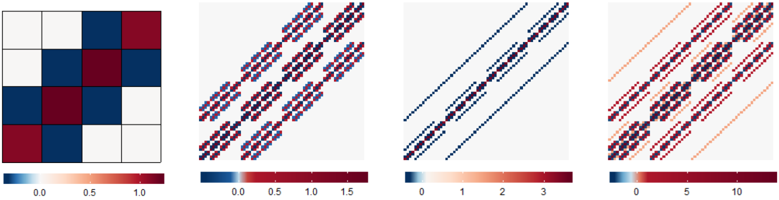
$$\mathbf{AZ} + \mathbf{ZB} = \mathbf{W}$$

SyGlasso exploits the Cartesian structure of differential operators[8], e.g., the Laplacian operator:

$$\nabla^2 u = (D_{xx} \oplus D_{yy})u \quad \Leftrightarrow \quad (\mathbf{D}_{xx} \oplus \mathbf{D}_{yy})\mathbf{u} \quad \Leftrightarrow \quad \mathbf{D}_{xx}\mathbf{U} + \mathbf{U}\mathbf{D}_{yy}$$

[8] Kressner, Tobler, *Krylov Subspace Methods for Linear Systems with Tensor Product Structure*. SIAM Journal on Matrix Analysis and Application, 2010

SyGlasso: Comparison of the Precision Matrix Structure

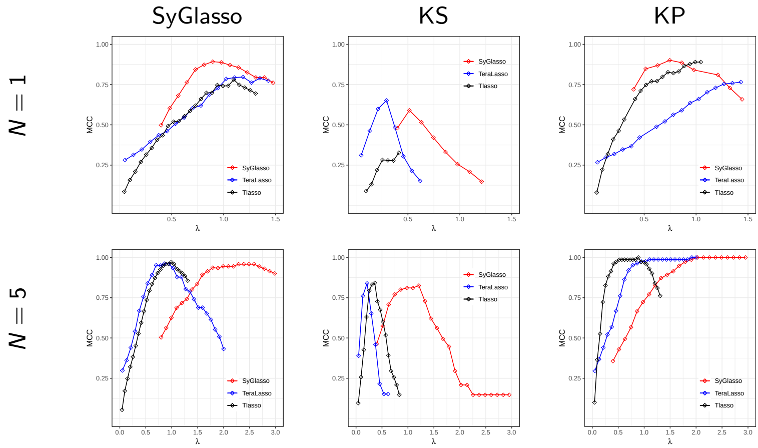


(a) $\boldsymbol{\Psi}_k$ (b) KP $\boldsymbol{\Omega}$ (c) KS $\boldsymbol{\Omega}$ (d) SyGlasso $\boldsymbol{\Omega}$

Note:

$$(\mathbf{A} \oplus \mathbf{B})^2 = \mathbf{A}^2 \oplus \mathbf{B}^2 + 2\mathbf{A} \otimes \mathbf{B}$$

SyGlasso: Recovery under Model Mismatch



$$\mathrm{MCC}^1 = \frac{TP \times TN - FP \times FN}{\sqrt{(TP+FP)(TP+FN)(TN+FP)(TN+FN)}}, \qquad \mathrm{MCC} = \begin{cases} 1, & \text{Perfect recovery} \\ 0, & \text{No recovery} \end{cases}$$

[1] Matthews Correlation Coefficient

Application to solar active region data

Data: Videos of active regions 13 hours before a B class flare or M/X class flare occurs

Observations are multiway arrays $\mathbf{Z}_i$, $i = 1, \ldots, n$, with dimensions

- $d_{time} = 13$ (1 hr cadence)
- $d_{width} = 100$
- $d_{height} = 50$
- $d_{channel} = 7$ (3 HMI and 4 AIA channels)

Application to solar active region data

Data: Videos of active regions 13 hours before a B class flare or M/X class flare occurs

Observations are multiway arrays $\mathbf{Z}_i$, $i = 1, \ldots, n$, with dimensions

- $d_{time} = 13$ (1 hr cadence)
- $d_{width} = 100$
- $d_{height} = 50$
- $d_{channel} = 7$ (3 HMI and 4 AIA channels)

Goal: Accurately estimate inverse covariance matrix of $\mathbf{Z}$ under 4-way sparse Sylvester model

$$(\mathbf{A} \oplus \mathbf{B} \oplus \mathbf{C} \oplus \mathbf{D})\mathbf{z} = \mathbf{w}$$

Special case: multichannel MIE

$$(\mathbf{D}_t \oplus \mathbf{D}_{xx} \oplus \mathbf{D}_{yy} \oplus \mathbf{D}_c)\mathbf{z} = \mathbf{w}$$

Application to solar active region data

Data: Videos of active regions 13 hours before a B class flare or M/X class flare occurs

Observations are multiway arrays $\mathbf{Z}_i$, $i = 1, \ldots, n$, with dimensions

- $d_{time} = 13$ (1 hr cadence)
- $d_{width} = 100$
- $d_{height} = 50$
- $d_{channel} = 7$ (3 HMI and 4 AIA channels)

Goal: Accurately estimate inverse covariance matrix of $\mathbf{Z}$ under 4-way sparse Sylvester model

$$(\mathbf{A} \oplus \mathbf{B} \oplus \mathbf{C} \oplus \mathbf{D})\mathbf{z} = \mathbf{w}$$

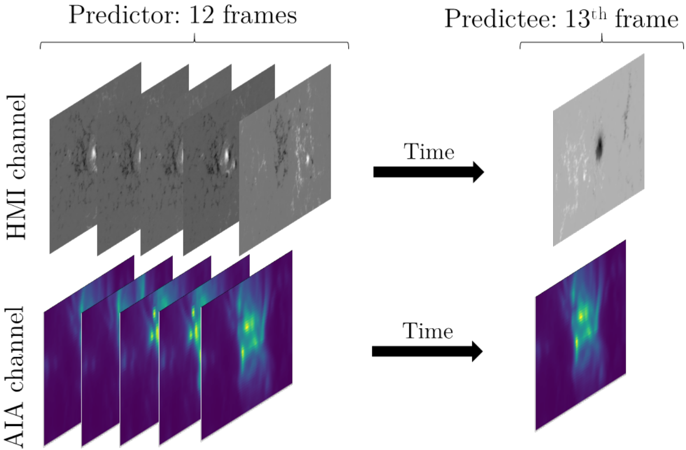
Special case: multichannel MIE

$$(\mathbf{D}_t \oplus \mathbf{D}_{xx} \oplus \mathbf{D}_{yy} \oplus \mathbf{D}_c)\mathbf{z} = \mathbf{w}$$

Model Validation

- Split dataset into a train and test set of videos.
- Train a one-step linear predictor to predict 13th frame from first 12 frames.
- Train QDA predictor to classify (B vs M/X) 13 frame from first 12 frames.
- Report performance of SyGlasso proximal alternating linearized minimization (SG-PALM) on the test set.

Application to solar active region data

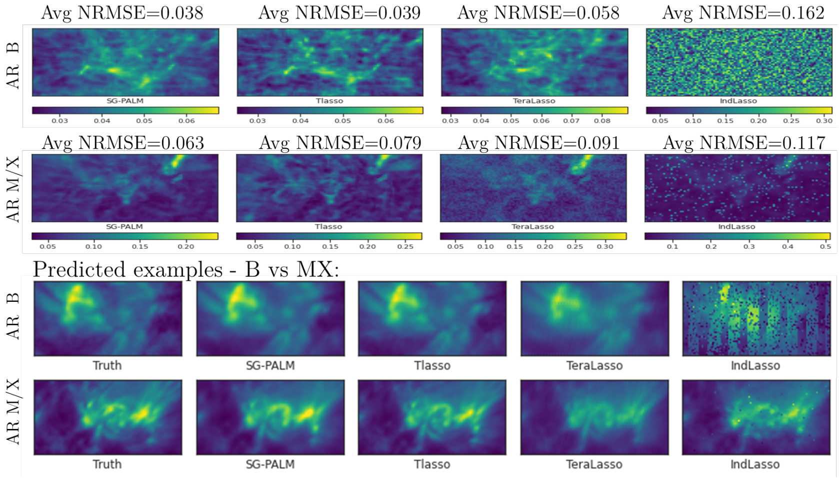


Active region images segmented by Xiantong Wang using whole disk data[9] [10]

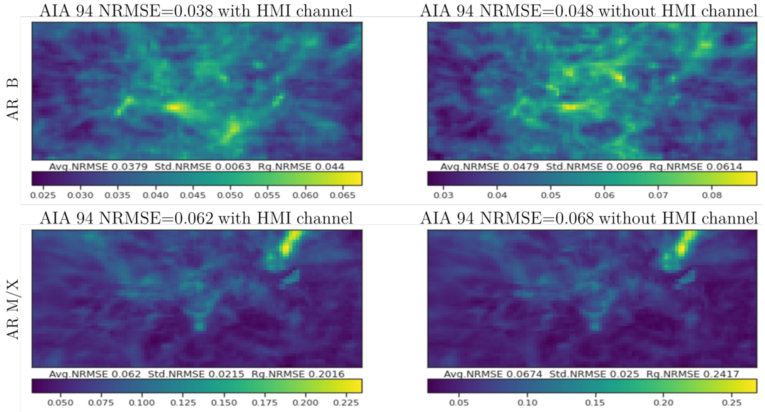
[9]Richard Galvez *et al*,*A Machine-learning Data Set Prepared from the NASA Solar Dynamics Observatory Mission*, The Astrophysical Journal, 2019

[10]Yu Wang and Alfred Hero, *A proximal alternating linearized minimization method for tensor graphical models*, submitted manuscript, Oct 2020.

Linear prediction of flare image from 12 hr pre-flare sequence

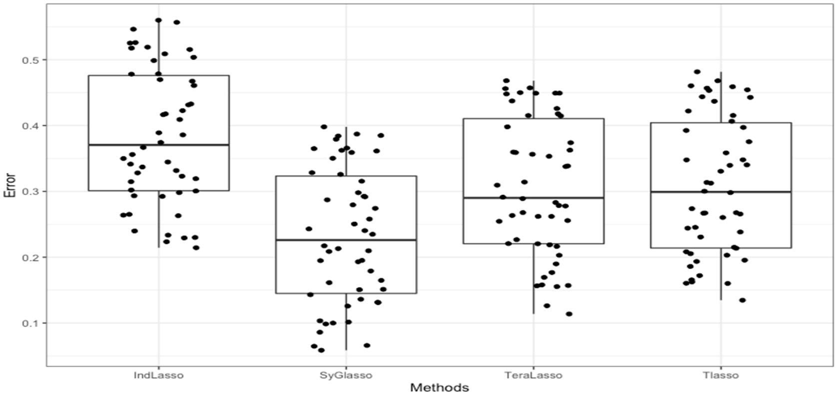


Value of HMI on AIA linear prediction accuracy



Main point: adding HMI channels to AIA improves prediction performance by factor > 20 for B class flares and > 9 for M/X class flares

QDA classifier of flare class B vs M/X from 12 hr pre-flare image sequence



Conclusions

Multiway covariance models that are motivated by mathematical physics

- Kronecker product PCA (KPCA)
    - Attains low rank representation of diffusion process covariance matrices

  Sparse Kronecker product model for inverse covariance $\boldsymbol{\Omega}$ (KLasso)
    - Reduces complexity of model from $O(p^2q^2)$ to $O(p+q)$
    - Bilinear non-convex objective function.
- Sparse Kronecker sum model for inverse covariance $\boldsymbol{\Omega}$ (TeraLasso)
    - Reduces complexity of model further than KLasso
    - Convex objective function
- Squared sparse Kronecker sum model for $\boldsymbol{\Omega}$ (SyGlasso)
    - KS square root factorization of $\boldsymbol{\Omega}$ equivalent to a Sylvester representation
    - Proximal alternating linearization maximization (PALM) implementation is fast and scalable