



POLYTECHNIQUE
MONTRÉAL

UNIVERSITÉ
D'INGÉNIERIE

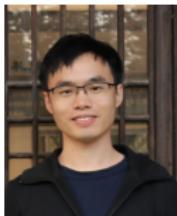


Low-Rank Matrix and Tensor Methods for Spatiotemporal Data Modeling

Xinyu Chen

University of Montreal, Canada

April 21, 2023



Ph.D. Candidate
Xinyu Chen



Supervisor
Prof. Nicolas Saunier



Co-supervisor
Prof. Lijun Sun

Current works:

- ① X. Chen, C. Zhang, X. Chen, N. Saunier, L. Sun (2022). Discovering dynamic patterns from spatiotemporal data with time-varying low-rank autoregression. arXiv preprint arXiv:2211.15482.
- ② X. Chen, Z. Cheng, N. Saunier, L. Sun (2022). Laplacian convolutional representation for traffic time series imputation. arXiv preprint arXiv:2212.01529.

GitHub repositories:

- ① **transdim**: Machine learning for spatiotemporal traffic data imputation and forecasting. (950 stars & 270 forks on GitHub)
<https://github.com/xinychen/transdim>
- ② **awesome-latex-drawing**: Academic drawing examples in LaTeX. (1,000 stars & 140 forks on GitHub)
<https://github.com/xinychen/awesome-latex-drawing>

Slides:

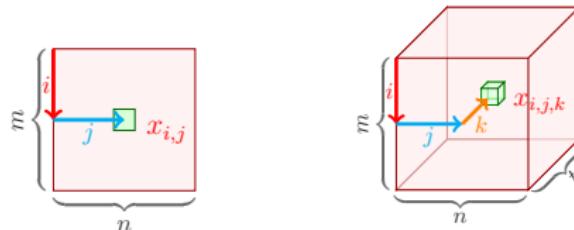
- https://xinychen.github.io/slides/stdata_modeling.pdf

Outline

- **Foundation: Tensor & Tensor Factorization**
 - What are Tensors?
 - Tensor Factorization (TF)
- **Time-Varying Autoregression**
 - Parameterize Coefficients via TF
 - Fluid Flow Data
 - Sea Surface Temperature Data
 - NYC Taxi Data
- **Low-Rank Laplacian Convolutional Model**
 - Reformulate Laplacian Regularization
 - Traffic Time Series Imputation
 - Multivariate Model for Speed Field Reconstruction
- **Matrix/Tensor Factorization**
 - Matrix Factorization
 - Smoothing Matrix Factorization
 - Hankel Tensor and Its Factorization
 - Spatiotemporal Hankel Tensor Factorization
 - Which Model Is Better?
- **Conclusion**

Foundation: Tensor & Tensor Factorization

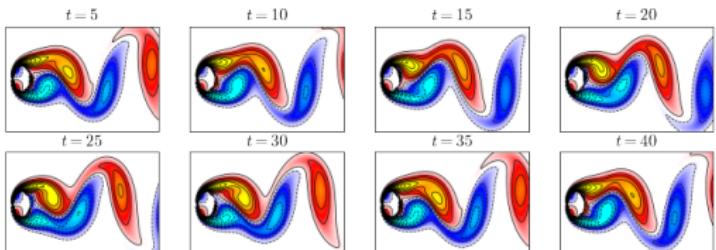
- What is tensor? $\mathbf{X} \in \mathbb{R}^{m \times n}$ vs. $\mathcal{X} \in \mathbb{R}^{m \times n \times t}$



- Tensors are everywhere!



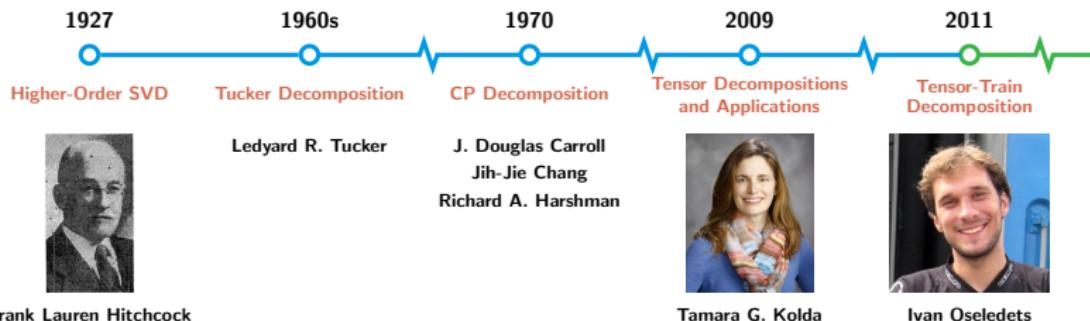
Color image with
RGB channels



Dynamical system (fluid flow)

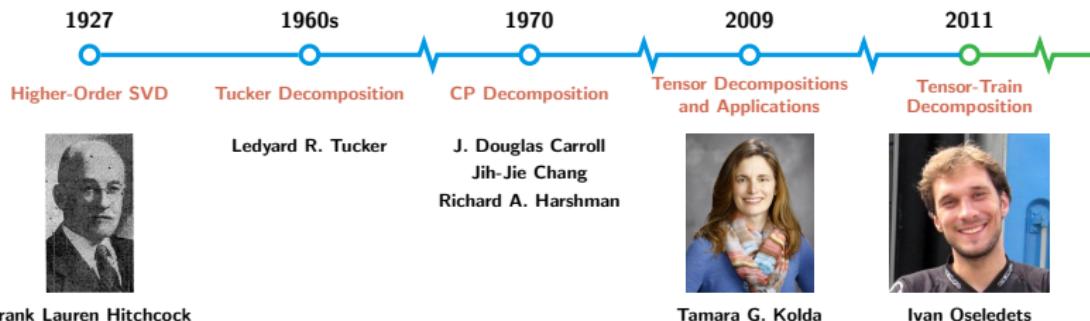
Foundation: Tensor & Tensor Factorization

- Revisit tensor factorization (TF)



Foundation: Tensor & Tensor Factorization

- Revisit tensor factorization (TF)

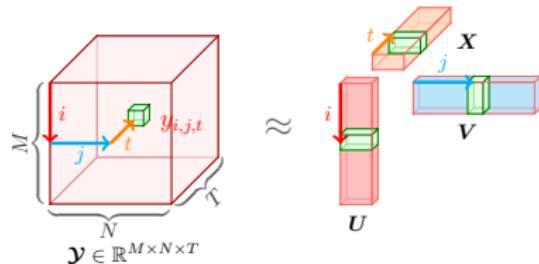


Frank Lauren Hitchcock

Tamara G. Kolda

Ivan Oseledets

- CP tensor factorization:** Factorize \mathcal{Y} into the combination of three rank- R factor matrices (i.e., low-dimensional latent factors).



$$\begin{cases} y_{i,j,t} \approx \sum_{r=1}^R u_{i,r} v_{j,r} x_{t,r} \\ \mathcal{Y} \approx \sum_{r=1}^R \mathbf{u}_r \otimes \mathbf{v}_r \otimes \mathbf{x}_r \end{cases}$$

Time-Varying Autoregression

- Given a sequence of spatiotemporal measurements

$$\mathbf{y}_t \in \mathbb{R}^N, t = 1, 2, \dots, T$$

$$\min_{\{\mathbf{A}_t\}} \underbrace{\frac{1}{2} \sum_t \|\mathbf{y}_t - \mathbf{A}_t \mathbf{y}_{t-1}\|_2^2}_{\text{Time-varying autoregression}}$$

[Over-parameterization] $\mathcal{O}(N^2(T-1))$ parameters vs. $\mathcal{O}(NT)$ data.

- (Ours)** Parameterize coefficients via TF¹:

$$\min_{\mathbf{W}, \mathbf{G}, \mathbf{V}, \mathbf{x}} \underbrace{\frac{1}{2} \sum_t \left\| \mathbf{y}_t - \mathbf{W} \mathbf{G} (\mathbf{x}_t^\top \otimes \mathbf{V})^\top \mathbf{y}_{t-1} \right\|_2^2}_{\text{Let } \mathbf{A}_t = \mathbf{G} \times_1 \mathbf{W} \times_2 \mathbf{V} \times_3 \mathbf{x}_t^\top \text{ be the TF}}$$

- Alternating minimization

$$\mathbf{W} := \{\mathbf{W} \mid \frac{\partial f}{\partial \mathbf{W}} = \mathbf{0}\} \quad \mathbf{G} := \{\mathbf{G} \mid \frac{\partial f}{\partial \mathbf{G}} = \mathbf{0}\}$$

$$\mathbf{V} := \{\mathbf{V} \mid \frac{\partial f}{\partial \mathbf{V}} = \mathbf{0}\} \quad \mathbf{x}_t := \{\mathbf{x}_t \mid \frac{\partial f}{\partial \mathbf{x}_t} = \mathbf{0}\}$$

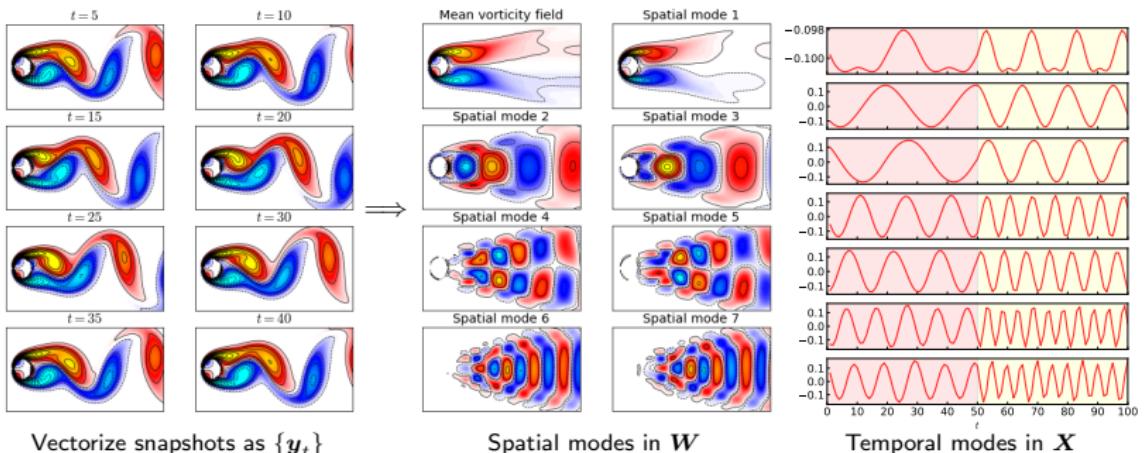
- Solve each subproblem by **conjugate gradient** or **least squares**.

¹X. Chen, C. Zhang, X. Chen, N. Saunier, L. Sun (2022). Discovering dynamic patterns from spatiotemporal data with time-varying low-rank autoregression. arXiv preprint arXiv:2211.15482.

- Time-varying autoregression with TF

$$\min_{\mathbf{W}, \mathbf{G}, \mathbf{V}, \mathbf{x}} \frac{1}{2} \sum_t \left\| \mathbf{y}_t - \mathbf{W} \mathbf{G} (\mathbf{x}_t^\top \otimes \mathbf{V})^\top \mathbf{y}_{t-1} \right\|_2^2$$

- **Fluid flow dataset** (the first 50 snapshots + 50 snapshots randomly selected from the last 100 snapshots)

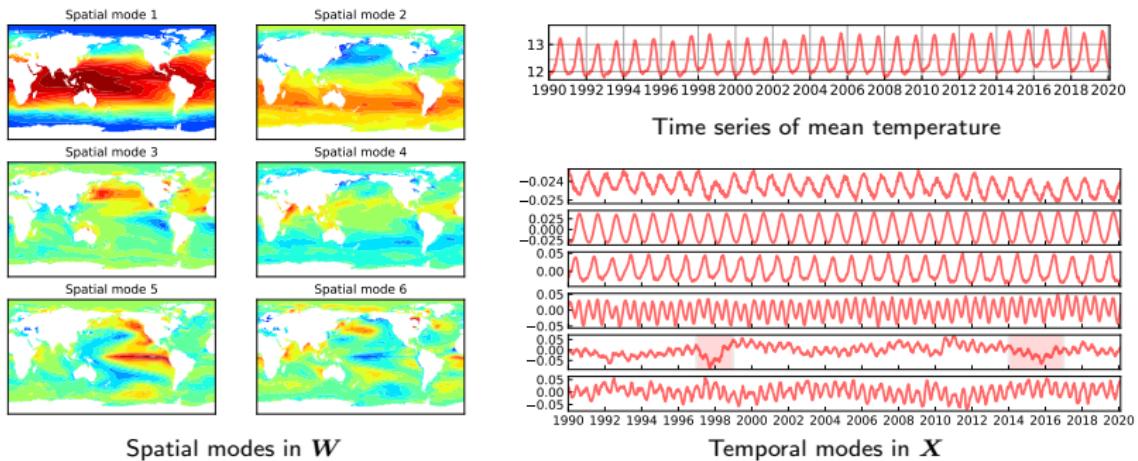


- Produce interpretable patterns and identify the system of different frequencies.

- Time-varying autoregression with TF

$$\min_{\mathbf{W}, \mathbf{G}, \mathbf{V}, \mathbf{x}} \frac{1}{2} \sum_t \left\| \mathbf{y}_t - \mathbf{W} \mathbf{G} (\mathbf{x}_t^\top \otimes \mathbf{V})^\top \mathbf{y}_{t-1} \right\|_2^2$$

- Sea surface temperature (**SST**) dataset

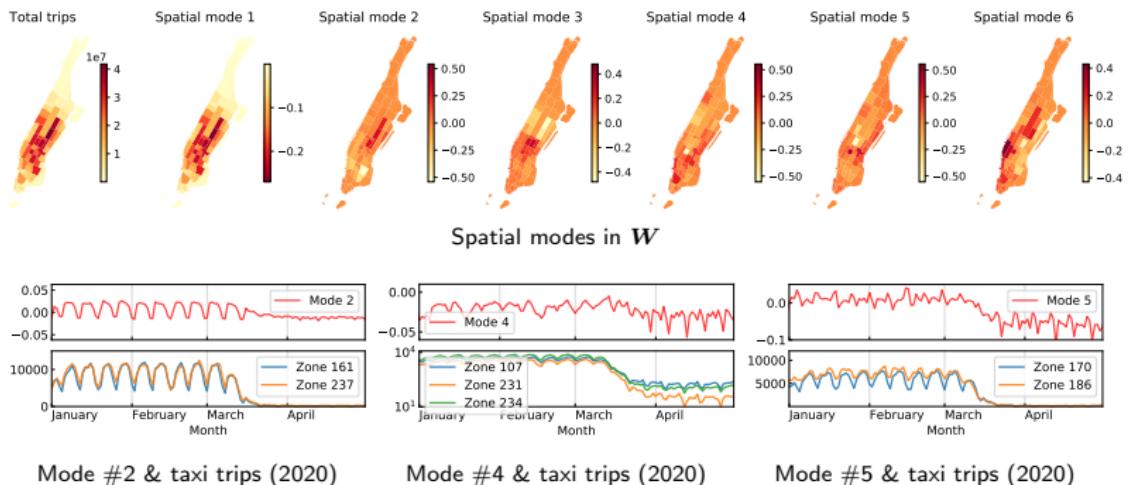


- Identify two strongest El Nino events (on 1997-98 & 2014-16).

- Time-varying autoregression with TF

$$\min_{\mathbf{W}, \mathbf{G}, \mathbf{V}, \mathbf{X}} \frac{1}{2} \sum_t \left\| \mathbf{y}_t - \mathbf{W} \mathbf{G} (\mathbf{x}_t^\top \otimes \mathbf{V})^\top \mathbf{y}_{t-1} \right\|_2^2$$

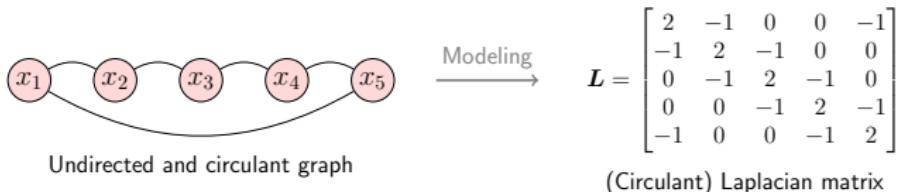
- NYC taxi dataset (pickup)



Low-Rank Laplacian Convolutional Model

Reformulate Laplacian regularization with circular convolution.

- Intuition of (circulant) Laplacian matrix.



- Laplacian kernel: $\ell = (2, -1, 0, 0, -1)^\top$.
- Define Laplacian kernel²:

$$\ell \triangleq (\underbrace{2\tau}_{\text{degree}}, \underbrace{-1, \dots, -1}_\tau, 0, \dots, 0, \underbrace{-1, \dots, -1}_\tau)^\top \in \mathbb{R}^T$$

for any time series $\mathbf{x} = (x_1, \dots, x_T)^\top \in \mathbb{R}^T$.

- Local trend modeling via (Laplacian) temporal regularization:

$$\mathcal{R}_\tau(\mathbf{x}) = \frac{1}{2} \|\mathbf{L}\mathbf{x}\|_2^2 = \frac{1}{2} \|\ell \star \mathbf{x}\|_2^2 = \frac{1}{2T} \|\mathcal{F}(\ell) \circ \mathcal{F}(\mathbf{x})\|_2^2$$

²X. Chen, Z. Cheng, N. Saunier, L. Sun (2022). Laplacian convolutional representation for traffic time series imputation. arXiv preprint arXiv:2212.01529.

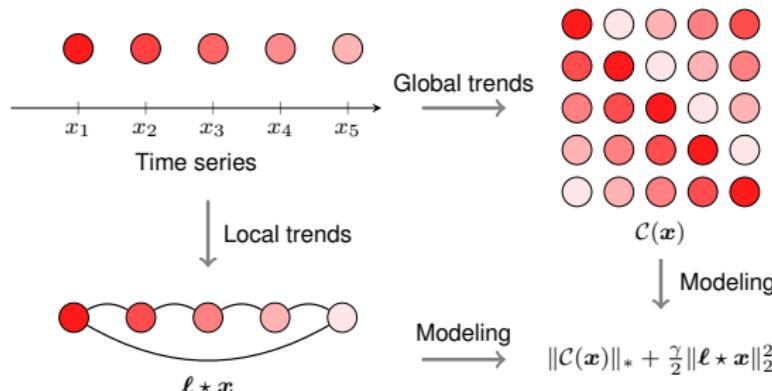
Low-Rank Laplacian Convolutional Model

Laplacian Convolutional Representation (LCR)

For any partially observed time series $\mathbf{y} \in \mathbb{R}^T$ with observed index set Ω , LCR utilizes circulant matrix and Laplacian kernel to characterize **global and local trends** in time series, respectively, i.e.,

$$\begin{aligned} & \min_{\mathbf{x}} \|\mathcal{C}(\mathbf{x})\|_* + \gamma \cdot \mathcal{R}_\tau(\mathbf{x}) \\ \text{s.t. } & \|\mathcal{P}_\Omega(\mathbf{x} - \mathbf{y})\|_2 \leq \epsilon \end{aligned}$$

where $\mathcal{C} : \mathbb{R}^T \rightarrow \mathbb{R}^{T \times T}$ denotes the circulant operator. $\|\cdot\|_*$ denotes the nuclear norm of matrix, namely, the sum of singular values.



Low-Rank Laplacian Convolutional Model

- LCR model:

$$\begin{aligned} \min_{\mathbf{x}} \quad & \|\mathcal{C}(\mathbf{x})\|_* + \gamma \cdot \mathcal{R}_\tau(\mathbf{x}) \\ \text{s.t.} \quad & \|\mathcal{P}_\Omega(\mathbf{x} - \mathbf{y})\|_2 \leq \epsilon \end{aligned}$$

- Augmented Lagrangian function:

$$\mathcal{L}(\mathbf{x}, \mathbf{z}, \mathbf{w}) = \|\mathcal{C}(\mathbf{x})\|_* + \frac{\gamma}{2} \|\ell * \mathbf{x}\|_2^2 + \frac{\lambda}{2} \|\mathbf{x} - \mathbf{z}\|_2^2 + \langle \mathbf{w}, \mathbf{x} - \mathbf{z} \rangle + \frac{\eta}{2} \|\mathcal{P}_\Omega(\mathbf{z} - \mathbf{y})\|_2^2$$

where $\mathbf{w} \in \mathbb{R}^T$ is the Lagrange multiplier, and $\langle \cdot, \cdot \rangle$ denotes the inner product.

- The ADMM scheme:

$$\left\{ \begin{array}{l} \mathbf{x} := \arg \min_{\mathbf{x}} \mathcal{L}(\mathbf{x}, \mathbf{z}, \mathbf{w}) \\ \mathbf{z} := \arg \min_{\mathbf{z}} \mathcal{L}(\mathbf{x}, \mathbf{z}, \mathbf{w}) \\ = \frac{1}{\lambda + \eta} \mathcal{P}_\Omega(\lambda \mathbf{x} + \mathbf{w} + \eta \mathbf{y}) + \mathcal{P}_\Omega^\perp(\mathbf{x} + \mathbf{w}/\lambda) \\ \mathbf{w} := \mathbf{w} + \lambda(\mathbf{x} - \mathbf{z}) \end{array} \right.$$

Low-Rank Laplacian Convolutional Model

- Optimize \mathbf{x} via fast Fourier transform (in $\mathcal{O}(T \log T)$ time):

$$\begin{aligned}\mathbf{x} &:= \arg \min_{\mathbf{x}} \|\mathcal{C}(\mathbf{x})\|_* + \frac{\gamma}{2} \|\ell * \mathbf{x}\|_2^2 + \frac{\lambda}{2} \|\mathbf{x} - \mathbf{z} + \mathbf{w}/\lambda\|_2^2 \\ \implies \hat{\mathbf{x}} &:= \arg \min_{\hat{\mathbf{x}}} \|\hat{\mathbf{x}}\|_1 + \frac{\gamma}{2T} \|\hat{\ell} \circ \hat{\mathbf{x}}\|_2^2 + \frac{\lambda}{2T} \|\hat{\mathbf{x}} - \hat{\mathbf{z}} + \hat{\mathbf{w}}/\lambda\|_2^2\end{aligned}$$

where we introduce $\{\hat{\ell}, \hat{\mathbf{x}}, \hat{\mathbf{z}}, \hat{\mathbf{w}}\}$ referring to $\{\ell, \mathbf{x}, \mathbf{z}, \mathbf{w}\}$ in the frequency domain.

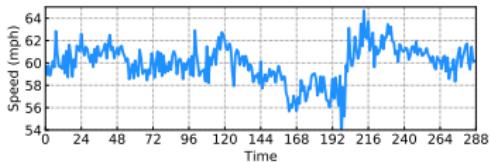
ℓ_1 -norm Minimization in Complex Space (Liu & Zhang'22)

For any optimization problem in the form of ℓ_1 -norm minimization in complex space:

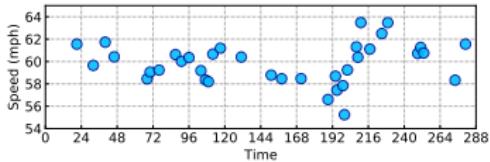
$$\min_{\hat{\mathbf{x}}} \|\hat{\mathbf{x}}\|_1 + \frac{\omega}{2} \|\hat{\mathbf{x}} - \hat{\mathbf{h}}\|_2^2$$

with complex-valued $\hat{\mathbf{x}}, \hat{\mathbf{h}} \in \mathbb{C}^T$, element-wise, the solution is given by

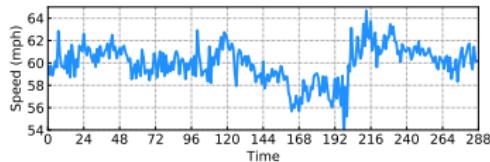
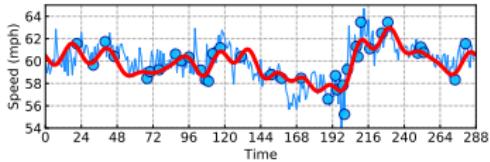
$$\hat{x}_t := \frac{\hat{h}_t}{|\hat{h}_t|} \cdot \max\{0, |\hat{h}_t| - 1/\omega\}, t = 1, \dots, T.$$



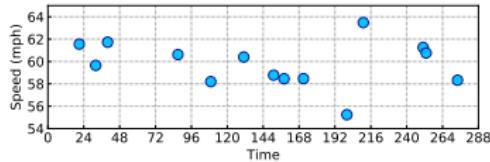
↓ Mask 90% observations



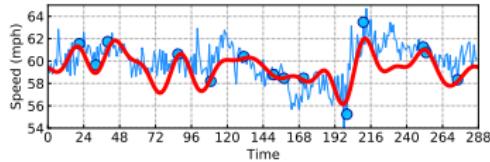
↓ Reconstruct time series



↓ Mask 95% observations



↓ Reconstruct time series

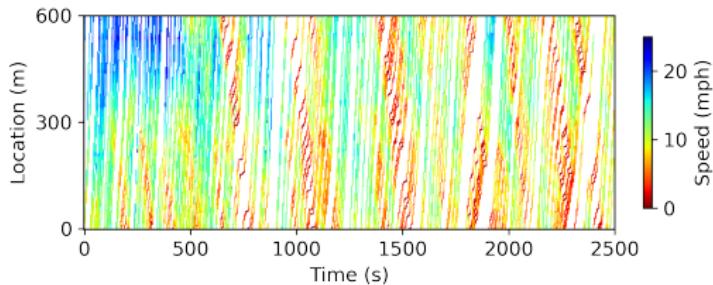


Multivariate LCR (LCR-2D)

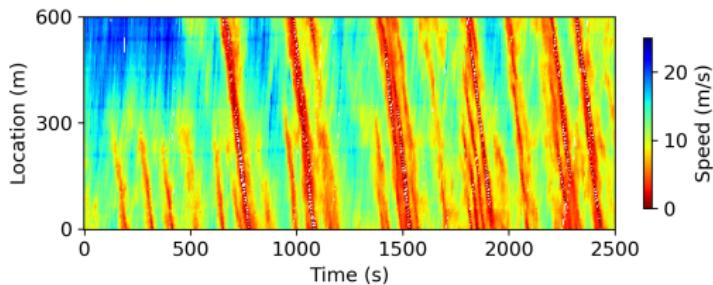
For any partially observed time series $\mathbf{Y} \in \mathbb{R}^{N \times T}$ with observed index set Ω , LCR can be formulated as follows,

$$\begin{aligned} \min_{\mathbf{X}} \quad & \|\mathcal{C}(\mathbf{X})\|_* + \frac{\gamma}{2} \|(\boldsymbol{\ell}_s \boldsymbol{\ell}^\top) \star \mathbf{X}\|_F^2 \\ \text{s.t. } & \|\mathcal{P}_\Omega(\mathbf{X} - \mathbf{Y})\|_F \leq \epsilon \end{aligned}$$

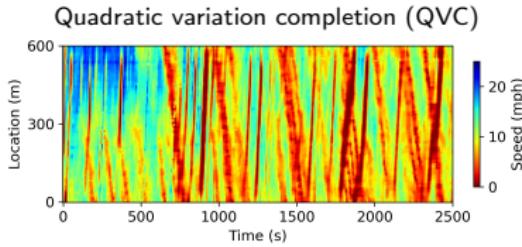
where $\mathcal{C} : \mathbb{R}^{N \times T} \rightarrow \mathbb{R}^{N \times N \times T \times T}$ denotes the circulant operator.



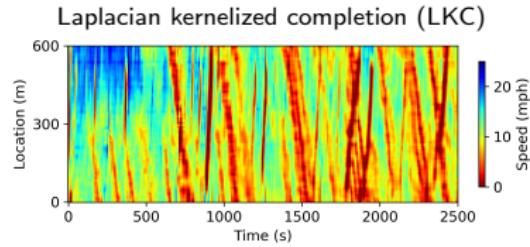
200-by-500 matrix
(NGSIM) ↓ Reconstruct speed field from
 20% sparse trajectories?



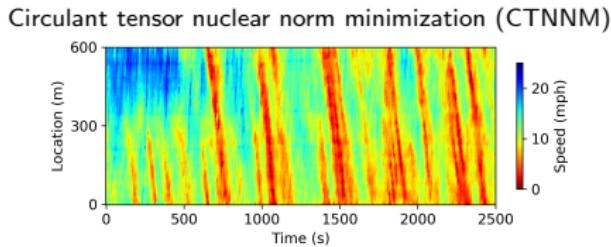
- How to learn from sparse spatiotemporal data?
- How to characterize spatial/temporal local dependencies?



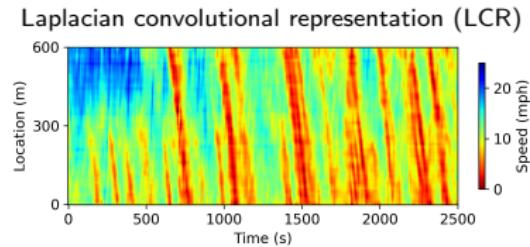
MAPE = 51.50% & RMSE = 4.86mph



MAPE = 46.94% & RMSE = 4.34mph



MAPE = 43.51% & RMSE = 1.65mph



MAPE = 41.29% & RMSE = 1.55mph

- QVC & LKC:

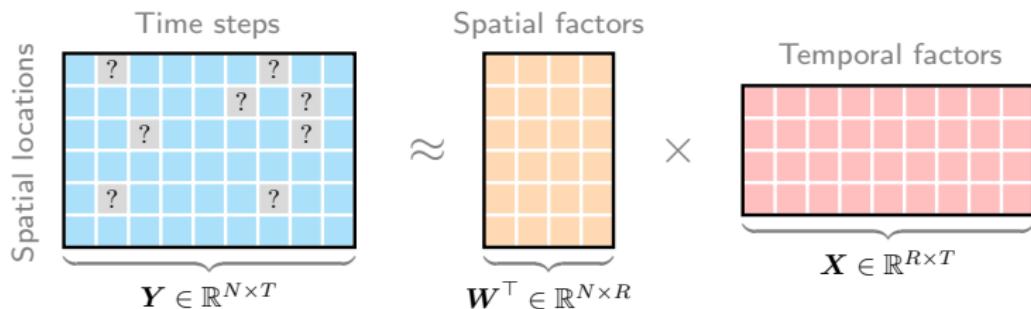
$$\begin{aligned} \min_{\mathbf{X}} \quad & \frac{\gamma}{2} \|(\ell_s \ell^\top) \star \mathbf{X}\|_F \\ \text{s.t. } & \|\mathcal{P}_\Omega(\mathbf{X} - \mathbf{Y})\|_F \leq \epsilon \end{aligned}$$

- CTNNM:

$$\begin{aligned} \min_{\mathbf{X}} \quad & \|\mathcal{C}(\mathbf{X})\|_* \\ \text{s.t. } & \|\mathcal{P}_\Omega(\mathbf{X} - \mathbf{Y})\|_F \leq \epsilon \end{aligned}$$

Matrix/Tensor Factorization

- Spatiotemporal data can be reconstructed by low-dimensional latent factors!



- MF optimization problem

$$\min_{\mathbf{W}, \mathbf{X}} \frac{1}{2} \left\| \mathcal{P}_\Omega(\mathbf{Y} - \mathbf{W}^\top \mathbf{X}) \right\|_F^2 + \frac{\rho}{2} (\|\mathbf{W}\|_F^2 + \|\mathbf{X}\|_F^2)$$

with factor matrices \mathbf{W} and \mathbf{X} . ($\|\cdot\|_F^2$ is the squared Frobenius norm.)

- Object function $f(\mathbf{W}, \mathbf{X})$ or f ;
- Rank $R \in \mathbb{N}^+$ ($R < \min\{N, T\}$);
- Orthogonal projection $\mathcal{P}_\Omega(\cdot)$.

Matrix/Tensor Factorization

- MF optimization problem

$$\min_{\mathbf{W}, \mathbf{X}} \frac{1}{2} \left\| \mathcal{P}_\Omega(\mathbf{Y} - \mathbf{W}^\top \mathbf{X}) \right\|_F^2 + \frac{\rho}{2} (\|\mathbf{W}\|_F^2 + \|\mathbf{X}\|_F^2)$$

- Partial derivatives

$$\begin{cases} \frac{\partial f}{\partial \mathbf{W}} = -\mathbf{X} \mathcal{P}_\Omega^\top (\mathbf{Y} - \mathbf{W}^\top \mathbf{X}) + \rho \mathbf{W} \\ \frac{\partial f}{\partial \mathbf{X}} = -\mathbf{W} \mathcal{P}_\Omega (\mathbf{Y} - \mathbf{W}^\top \mathbf{X}) + \rho \mathbf{X} \end{cases}$$

- Alternating least squares (**ALS**)

$$\begin{cases} \frac{\partial f}{\partial \mathbf{W}} = \mathbf{0} \\ \frac{\partial f}{\partial \mathbf{X}} = \mathbf{0} \end{cases} \implies \begin{cases} \mathbf{w}_i := \left(\sum_{t:(i,t) \in \Omega} \mathbf{x}_t \mathbf{x}_t^\top + \rho \mathbf{I}_R \right)^{-1} \sum_{t:(i,t) \in \Omega} \mathbf{x}_t y_{i,t} \\ \mathbf{x}_t := \left(\sum_{i:(i,t) \in \Omega} \mathbf{w}_i \mathbf{w}_i^\top + \rho \mathbf{I}_R \right)^{-1} \sum_{i:(i,t) \in \Omega} \mathbf{w}_i y_{i,t} \end{cases}$$

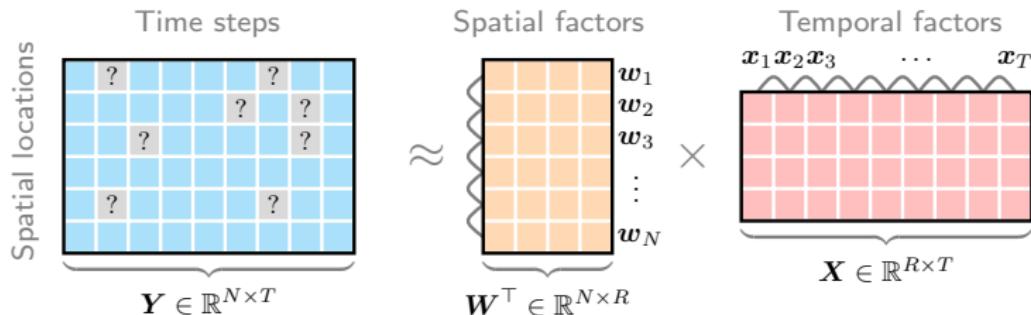
- Latent factors

- $\mathbf{w}_i \in \mathbb{R}^R$, $i = 1, 2, \dots, N$ are the columns of \mathbf{W} ;
- $\mathbf{x}_t \in \mathbb{R}^R$, $t = 1, 2, \dots, T$ are the columns of \mathbf{X} .

Matrix/Tensor Factorization

Smoothing matrix factorization

- Spatial/temporal local dependencies are also important!



- Formulate spatial/temporal dependencies

$$\mathbf{W}\Psi_1^\top = \begin{bmatrix} & & & \\ \mathbf{w}_2 - \mathbf{w}_1 & \cdots & \mathbf{w}_N - \mathbf{w}_{N-1} & \\ & & & \end{bmatrix}$$
$$\mathbf{X}\Psi_2^\top = \begin{bmatrix} & & & \\ \mathbf{x}_2 - \mathbf{x}_1 & \cdots & \mathbf{x}_T - \mathbf{x}_{T-1} & \\ & & & \end{bmatrix}$$

Matrix/Tensor Factorization

Smoothing Matrix Factorization

- Formulate spatial/temporal dependencies

$$\Psi = \begin{bmatrix} -1 & 1 & 0 & \cdots & 0 & 0 \\ 0 & -1 & 1 & \cdots & 0 & 0 \\ 0 & 0 & -1 & \cdots & 0 & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & 0 & \cdots & -1 & 1 \end{bmatrix} \implies \begin{cases} \|\mathbf{W}\Psi_1^\top\|_F^2 & \text{with } \Psi_1 \in \mathbb{R}^{(N-1) \times N} \\ \|\mathbf{X}\Psi_2^\top\|_F^2 & \text{with } \Psi_2 \in \mathbb{R}^{(T-1) \times T} \end{cases}$$

- SMF optimization problem

$$\begin{aligned} \min_{\mathbf{W}, \mathbf{X}} \quad & \frac{1}{2} \left\| \mathcal{P}_\Omega(\mathbf{Y} - \mathbf{W}^\top \mathbf{X}) \right\|_F^2 + \frac{\rho}{2} (\|\mathbf{W}\|_F^2 + \|\mathbf{X}\|_F^2) \\ & + \frac{\lambda}{2} (\|\mathbf{W}\Psi_1^\top\|_F^2 + \|\mathbf{X}\Psi_2^\top\|_F^2) \end{aligned}$$

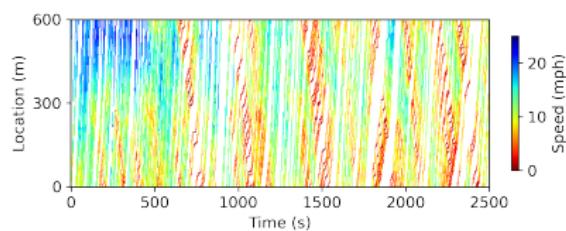
- Alternating minimization

$$\mathbf{W} := \{ \mathbf{W} \mid \frac{\partial f}{\partial \mathbf{W}} = \mathbf{0} \} \quad \mathbf{X} := \{ \mathbf{X} \mid \frac{\partial f}{\partial \mathbf{X}} = \mathbf{0} \}$$

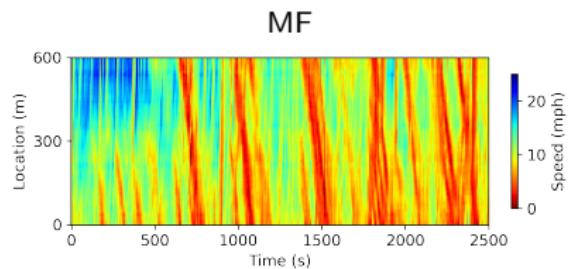
- Solve each matrix equation by the **conjugate gradient** method.

Matrix/Tensor Factorization

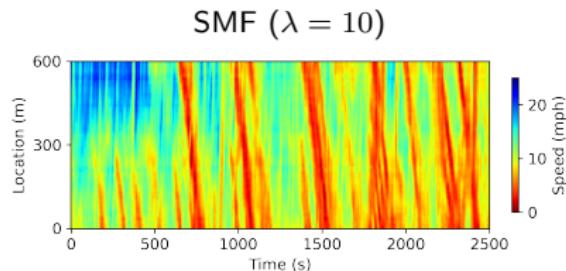
- Speed field reconstruction
 - Set rank $R = 10$, weight parameter $\rho = 10$.



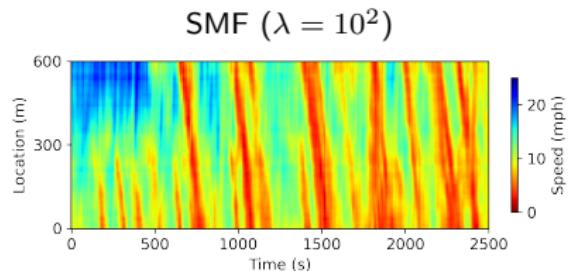
Sparse speed field



MF
MAPE = 45.84%, RMSE = 2.80mph



SMF ($\lambda = 10$)
MAPE = 44.06%, RMSE = 2.16mph



SMF ($\lambda = 10^2$)
MAPE = 48.00%, RMSE = 1.60mph

Matrix/Tensor Factorization

Hankel tensor and its factorization

- Hankel matrix

- Given $\mathbf{y} = (1, 2, 3, 4, 5)^\top$ and window length $\tau = 2$, we have

$$\mathcal{H}_\tau(\mathbf{y}) = \begin{bmatrix} 1 & 2 \\ 2 & 3 \\ 3 & 4 \\ 4 & 5 \end{bmatrix} \in \mathbb{R}^{4 \times 2}$$

- On time series $\mathbf{y} = (y_1, y_2, \dots, y_5)^\top$ with $\tau = 2$:

$$\mathcal{H}_\tau(\mathbf{y}) = \begin{bmatrix} y_1 & y_2 \\ y_2 & y_3 \\ y_3 & y_4 \\ y_4 & y_5 \end{bmatrix} \approx \begin{bmatrix} v_1 \\ v_2 \\ v_3 \\ v_4 \end{bmatrix} \otimes \begin{bmatrix} x_1 \\ x_2 \end{bmatrix}$$

$$\implies \hat{\mathbf{y}} = \begin{bmatrix} \hat{y}_1 \\ \hat{y}_2 \\ \hat{y}_3 \\ \hat{y}_4 \\ \hat{y}_5 \end{bmatrix} = \mathcal{H}_\tau^{-1} \left(\begin{bmatrix} v_1 x_1 & v_1 x_2 \\ v_2 x_1 & v_2 x_2 \\ v_3 x_1 & v_3 x_2 \\ v_4 x_1 & v_4 x_2 \end{bmatrix} \right) = \begin{bmatrix} v_1 x_1 \\ (v_1 x_2 + v_2 x_1)/2 \\ (v_2 x_2 + v_3 x_1)/2 \\ (v_3 x_2 + v_4 x_1)/2 \\ v_4 x_2 \end{bmatrix}$$

- Automatic temporal modeling.

Matrix/Tensor Factorization

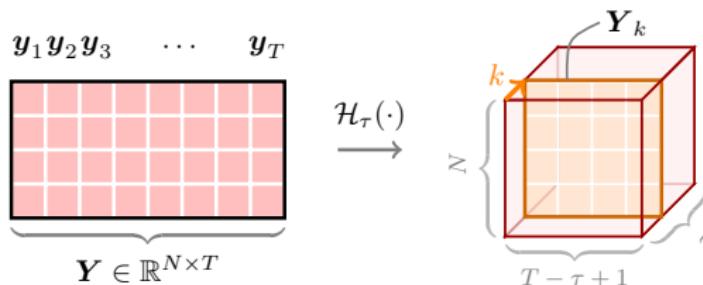
Hankel tensor and its factorization

- (Hankelization) Hankel tensor $\mathcal{H}_\tau(\mathbf{Y})$

- Tensor size: $N \times (T - \tau + 1) \times \tau$;

- Slices: $\mathbf{Y}_k = \begin{bmatrix} | & | & & | \\ \mathbf{y}_k & \mathbf{y}_{k+1} & \cdots & \mathbf{y}_{T-\tau+k} \\ | & | & & | \end{bmatrix}, k = 1, 2, \dots, \tau$;

- Slice size: $N \times (T - \tau + 1)$.



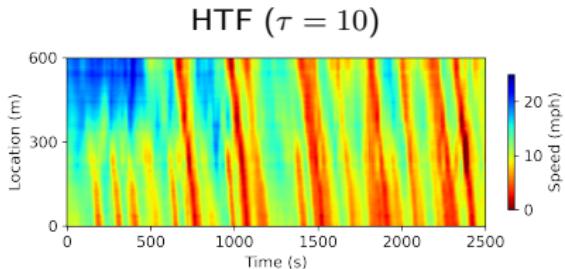
Matrix/Tensor Factorization

Hankel tensor and its factorization

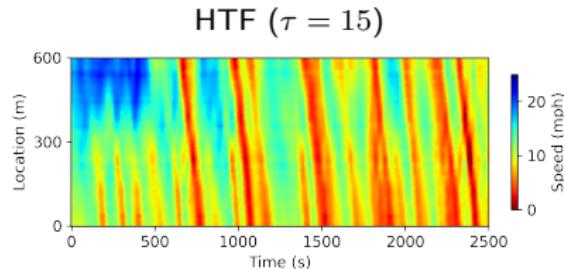
- HTF optimization problem

$$\min_{\mathbf{U}, \mathbf{V}, \mathbf{X}} \frac{1}{2} \left\| \mathcal{P}_{\tilde{\Omega}} \left(\mathcal{H}_\tau(\mathbf{Y}) - \sum_{r=1}^R \mathbf{u}_r \otimes \mathbf{v}_r \otimes \mathbf{x}_r \right) \right\|_F^2$$

- HTF's advantage/disadvantage over MF:
 - ✓ Automatic temporal modeling ✗ High memory consumption
- Speed field reconstruction
 - Set rank $R = 10$;
 - Recall that SMF: MAPE = 48.00% & RMSE = 1.60mph.



MAPE = 41.40%, RMSE = 1.42mph

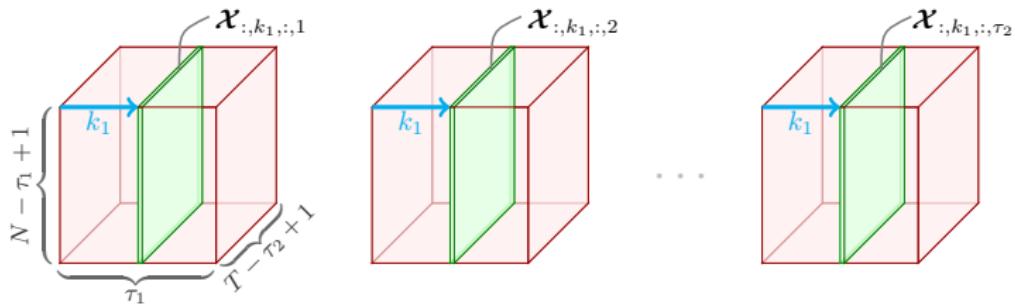


MAPE = 43.97%, RMSE = 1.42mph

Matrix/Tensor Factorization

Spatiotemporal Hankel tensor factorization

- Hankelization from $\mathbf{X} \in \mathbb{R}^{N \times T}$ to $\mathcal{X} \triangleq \mathcal{H}_{\tau_1, \tau_2}(\mathbf{X})$ (Hankel tensor).
 - Tensor size: $(N - \tau_1 + 1) \times \tau_1 \times (T - \tau_2 + 1) \times \tau_2$;
 - Slice: $\mathcal{X}_{:, k_1, :, k_2}, \forall k_1, k_2$;
 - Slice size: $(N - \tau_1 + 1) \times (T - \tau_2 + 1)$.



- StHTF optimization problem

$$\min_{\mathbf{Q}, \mathbf{S}, \mathbf{U}, \mathbf{V}} \frac{1}{2} \left\| \mathcal{P}_{\Omega} \left(\mathcal{H}_{\tau_1, \tau_2}(\mathbf{Y}) - \sum_{r=1}^R \mathbf{q}_r \otimes \mathbf{s}_r \otimes \mathbf{u}_r \otimes \mathbf{v}_r \right) \right\|_F^2$$

Matrix/Tensor Factorization

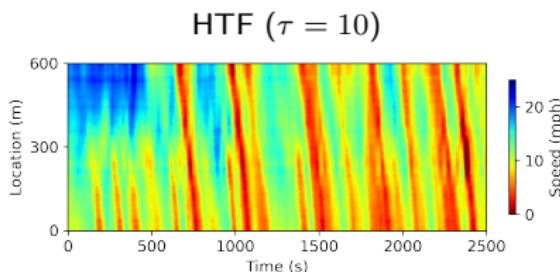
Spatiotemporal Hankel tensor factorization

- StHTF optimization problem

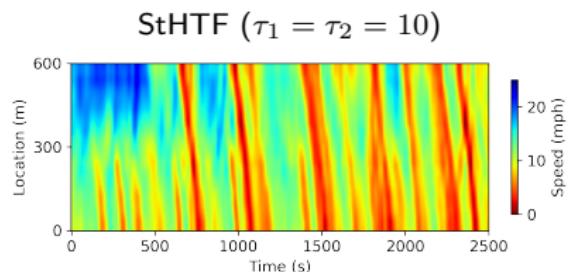
$$\min_{Q, S, U, V} \frac{1}{2} \left\| \mathcal{P}_{\Omega} \left(\mathcal{H}_{\tau_1, \tau_2}(Y) - \sum_{r=1}^R \mathbf{q}_r \otimes \mathbf{s}_r \otimes \mathbf{u}_r \otimes \mathbf{v}_r \right) \right\|_F^2$$

- Speed field reconstruction

- Set rank $R = 10$;
 - Recall that SMF: MAPE = 48.00% & RMSE = 1.60mph.



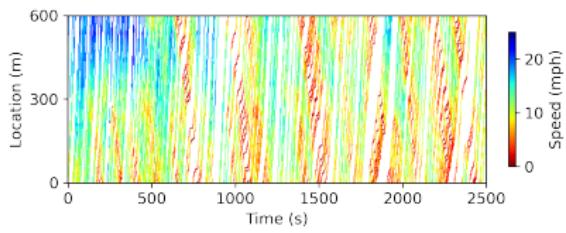
MAPE = 41.40%, RMSE = 1.42mph



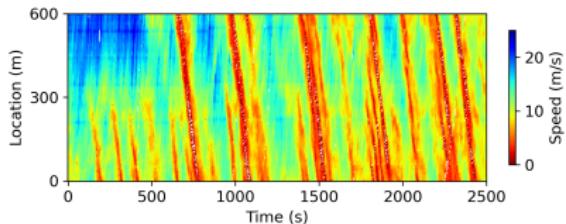
MAPE = 41.58%, RMSE = 1.39mph

Matrix/Tensor Factorization

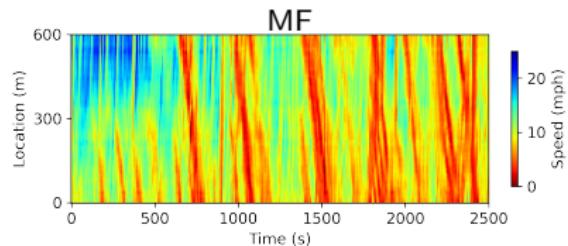
Which Model Is Better?



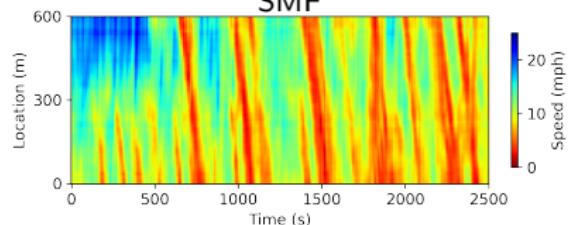
Sparse speed field



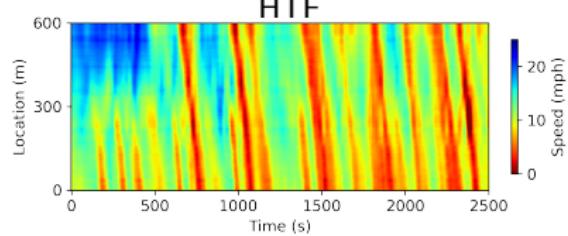
Ground truth speed field



MF



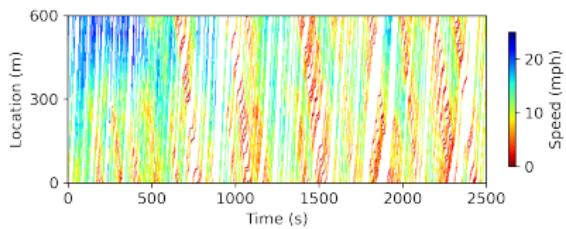
SMF



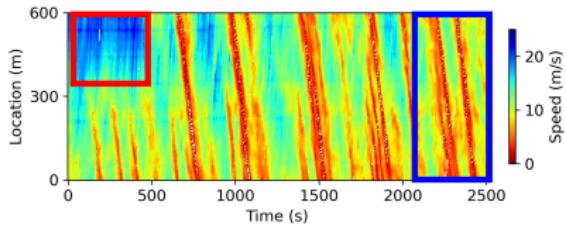
HTF

Matrix/Tensor Factorization

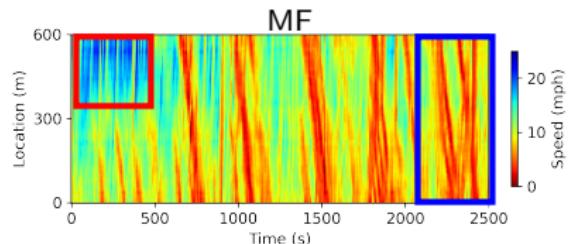
Which Model Is Better?



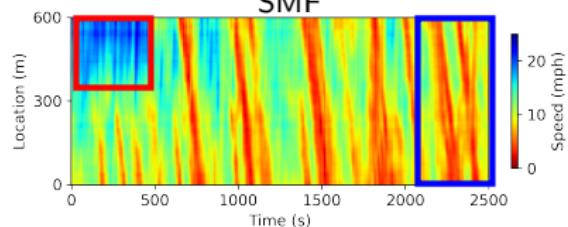
Sparse speed field



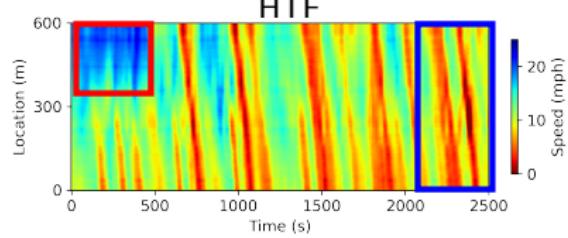
Ground truth speed field



MF



SMF



HTF

Conclusion

- Time-varying autoregression
 - use TF to compress the coefficients
 - interpret factorization structure as spatial/temporal modes
- Low-rank Laplacian convolutional representation
 - characterize local trends of time series via the Laplacian kernels
 - solve the model by a fast FFT implementation
- Spatiotemporal matrix/tensor factorization
 - Smoothing regularization
 - Hankelization



Thanks for your attention!

Any Questions?

About me:

- 🏠 Homepage: <https://xinychen.github.io>
- Γ Google Scholar: [user=mCrW04wAAAAJhl](#) (650+ citations)
- ⌚ GitHub: <https://github.com/xinychen> (3.2k+ stars)
- Ⓜ Blog: <https://medium.com/@xinyu.chen> (65k+ views)
- ✉ How to reach me: chenxy346@gmail.com