



MENS
MANUS AND
MACHINA

Modeling Urban Traffic Data with Matrix and Tensor Approaches

Xinyu Chen

Postdoctoral Associate, MIT

Ph.D., University of Montreal

October 21, 2024

Urban Traffic Data

- Transport & mobility application scenarios



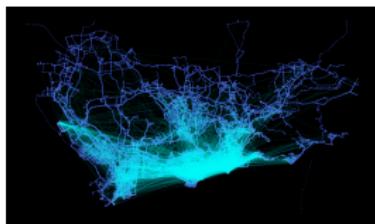
Highway (Portland)



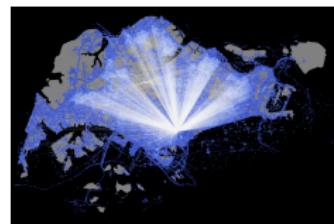
Uber movement (NYC)



Uber movement (Seattle)



Taxi trajectory (Shenzhen)



Passenger flow (Singapore)

- Challenges: Sparsity, time-varying system, high-dimensionality, and multi-dimensionality

Papers:

- X. Chen, Z. Cheng, H.Q. Cai, N. Saunier, L. Sun (2024). "Laplacian Convolutional Representation for Traffic Time Series Imputation". *IEEE Transactions on Knowledge and Data Engineering*, 36 (11): 6490–6502.
- X. Chen, L. Sun (2022). "Bayesian Temporal Factorization for Multidimensional Time Series Prediction". *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 44 (9): 4659–4673.
- X. Chen, X.L. Zhao, C. Cheng (2024). "Forecasting Urban Traffic States with Sparse Data Using Hankel Temporal Matrix Factorization". *INFORMS Journal on Computing*. Early access.
- X. Chen, C. Zhang, X. Chen, N. Saunier, L. Sun (2024). "Discovering Dynamic Patterns from Spatiotemporal Data with Time-Varying Low-Rank Autoregression". *IEEE Transactions on Knowledge and Data Engineering*, 36 (2): 504–517.

Papers:

- X. Chen, Z. Cheng, H.Q. Cai, N. Saunier, L. Sun (2024). "Laplacian Convolutional Representation for Traffic Time Series Imputation". IEEE Transactions on Knowledge and Data Engineering, 36 (11): 6490–6502.
- X. Chen, L. Sun (2022). "Bayesian Temporal Factorization for Multidimensional Time Series Prediction". IEEE Transactions on Pattern Analysis and Machine Intelligence, 44 (9): 4659–4673.
- X. Chen, X.L. Zhao, C. Cheng (2024). "Forecasting Urban Traffic States with Sparse Data Using Hankel Temporal Matrix Factorization". INFORMS Journal on Computing. Early access.
- X. Chen, C. Zhang, X. Chen, N. Saunier, L. Sun (2024). "Discovering Dynamic Patterns from Spatiotemporal Data with Time-Varying Low-Rank Autoregression". IEEE Transactions on Knowledge and Data Engineering, 36 (2): 504–517.

ML \Rightarrow Imputation & Prediction & Pattern Discovery

Laplacian Convolutional Representation for Traffic Time Series Imputation

IEEE Transactions on Knowledge and Data Engineering, 2024

<https://doi.org/10.1109/TKDE.2024.3419698>



Xinyu Chen



Zhanhong Cheng



HanQin Cai



Nicolas Saunier



Lijun Sun

Materials:

- GitHub: <https://github.com/xinyuchen/transdim>
- Blog: https://spatiotemporal-data.github.io/posts/ts_conv

Traffic Flow Data

- Portland highway traffic data¹



Highway network & sensor locations



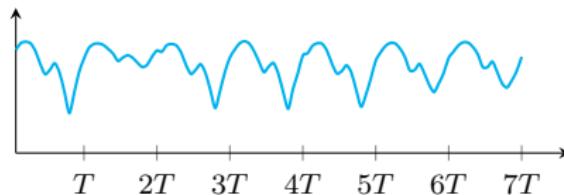
- $X \in \mathbb{R}^{N \times T}$ with N spatial locations $\times T$ time steps
 - Traffic volume/speed shows strong spatial/temporal dependencies

¹ <https://portal.its.pdx.edu/home>

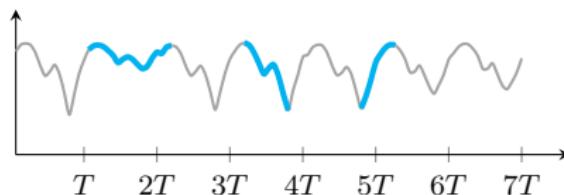
Time Series Imputation

Motivation: Traffic imputation

- Global trends (e.g., long-term quasi-seasonality & daily/weekly rhythm):



- Local trends (e.g., short-term time series trends):



How to characterize both global and local trends in sparse time series?

Local Trend Modeling

- Intuition of (circulant) Laplacian matrix

Undirected and circulant graph

Modeling

$$\mathbf{L} = \begin{bmatrix} 2 & -1 & 0 & 0 & -1 \\ -1 & 2 & -1 & 0 & 0 \\ 0 & -1 & 2 & -1 & 0 \\ 0 & 0 & -1 & 2 & -1 \\ -1 & 0 & 0 & -1 & 2 \end{bmatrix}$$

(Circulant) Laplacian matrix

- Define Laplacian kernel:

$$\boldsymbol{\ell} \triangleq (2, -1, 0, 0, -1)^\top$$

⇓

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

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

- Temporal regularization (w/ circular convolution \star):

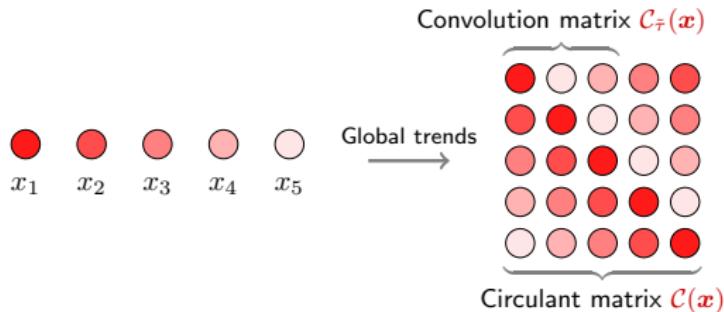
$$\mathcal{R}_\tau(\mathbf{x}) = \frac{1}{2} \|\mathbf{L}\mathbf{x}\|_2^2 = \frac{1}{2} \|\boldsymbol{\ell} \star \mathbf{x}\|_2^2$$

“... The circulant graph has an adjacency matrix that is a circulant matrix.”

— Circulant graph on Wikipedia

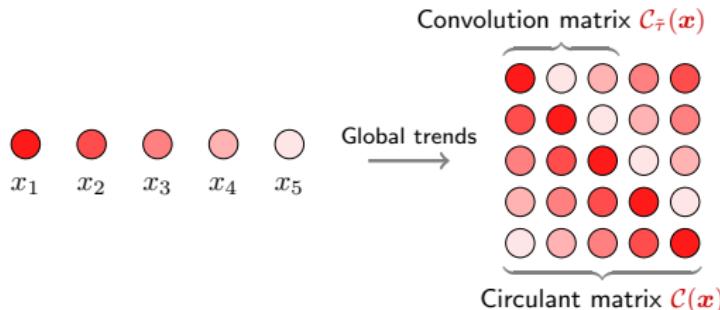
Global Trend Modeling

Circulant matrix $\mathcal{C}(\mathbf{x})$ vs. convolution matrix $\mathcal{C}_{\tilde{\tau}}(\mathbf{x})$



Global Trend Modeling

Circulant matrix $\mathcal{C}(\mathbf{x})$ vs. convolution matrix $\mathcal{C}_{\tilde{\tau}}(\mathbf{x})$



- Circulant/Convolution nuclear norm minimization
 - A balance between global and local trends modeling?

CircNNM (Liu'22, Liu & Zhang'23)

Estimating \mathbf{x} :

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

on data \mathbf{y} w/ observed index set Ω .

ConvNNM (Liu'22, Liu & Zhang'23)

Estimating \mathbf{x} :

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

on data \mathbf{y} w/ observed index set Ω .

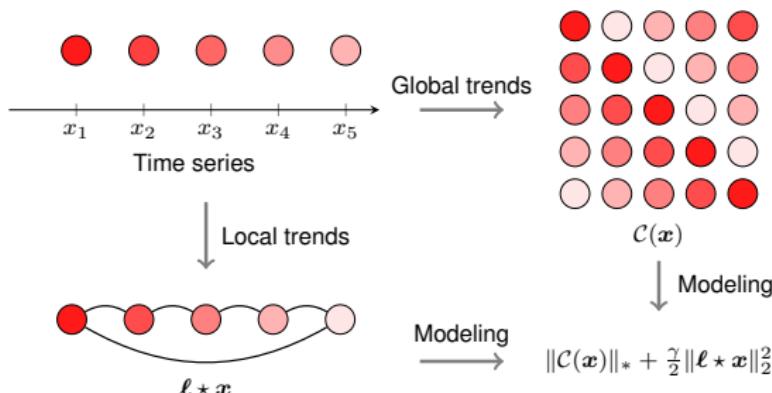
Global + Local Trends?

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.,

$$\min_{\mathbf{x}} \underbrace{\|\mathcal{C}(\mathbf{x})\|_*}_{\text{global}} + \frac{\gamma}{2} \underbrace{\|\ell * \mathbf{x}\|_2^2}_{\text{local}}$$

s.t. $\|\mathcal{P}_\Omega(\mathbf{x} - \mathbf{y})\|_2 \leq \epsilon$



Laplacian Convolutional Representation

- Augmented Lagrangian function:²

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

- The ADMM scheme:

$$\begin{cases} \mathbf{x} := \arg \min_{\mathbf{x}} \mathcal{L}(\mathbf{x}, \mathbf{z}, \mathbf{w}) & \text{(Nuclear norm minimization)} \\ \mathbf{z} := \arg \min_{\mathbf{z}} \mathcal{L}(\mathbf{x}, \mathbf{z}, \mathbf{w}) & \text{(Closed-form solution)} \\ \mathbf{w} := \mathbf{w} + \lambda(\mathbf{x} - \mathbf{z}) & \text{(Standard update)} \end{cases}$$

- Optimize \mathbf{x} ?

$$\underbrace{\|\mathcal{C}(\mathbf{x})\|_* = \|\mathcal{F}(\mathbf{x})\|_1}_{\text{property of circulant matrix}} \quad \& \quad \underbrace{\frac{1}{2} \|\ell * \mathbf{x}\|_2^2 = \frac{1}{2T} \|\mathcal{F}(\ell) \circ \mathcal{F}(\mathbf{x})\|_2^2}_{\text{property of circular convolution}}$$

Nuclear norm minimization \Rightarrow **ℓ_1 -norm minimization with FFT** in $\mathcal{O}(T \log T)$ time.

² $\mathbf{w} \in \mathbb{R}^T$ (Lagrange multiplier); $\langle \cdot, \cdot \rangle$ (inner product).

Laplacian Convolutional Representation

- Optimize \mathbf{x} via FFT (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 $\{\hat{\ell}, \hat{\mathbf{x}}, \hat{\mathbf{z}}, \hat{\mathbf{w}}\} \triangleq \mathcal{F}\{\ell, \mathbf{x}, \mathbf{z}, \mathbf{w}\}$ (i.e., FFT).

ℓ_1 -norm Minimization in Complex Space (Liu & Zhang'23)

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

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

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

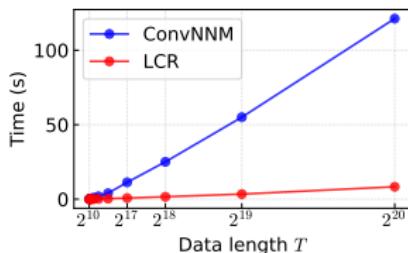
$$\hat{x}_t := \frac{\hat{h}_t}{|\hat{h}_t|} \cdot \max\{0, |\hat{h}_t| - 1/\delta\}, t \in [T].$$

Laplacian Convolutional Representation

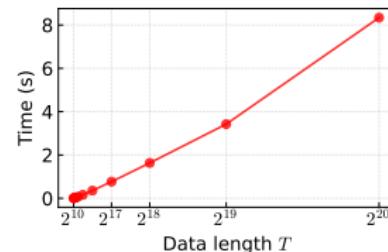
Empirical time complexity

On the synthetic data $\mathbf{y} \in \mathbb{R}^T$ with $T \in \{2^{10}, 2^{11}, \dots, 2^{20}\}$

- Ours: **LCR**
 - An FFT implementation in $\mathcal{O}(T \log T)$
 - The logarithmic factor $\log T$ makes the FFT highly efficient
- Baseline: **ConvNNM** (Liu'22, Liu & Zhang'23)
 - Convolution matrix $\mathcal{C}_{\tilde{\tau}}(\mathbf{y}) \in \mathbb{R}^{T \times \tilde{\tau}}$ with kernel size $\tilde{\tau} = 2^4$
 - Singular value thresholding in $\mathcal{O}(\tilde{\tau}^2 T)$

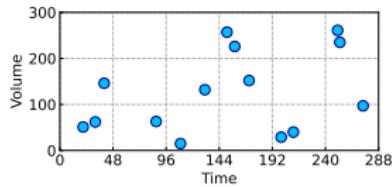


ConvNNM vs. LCR

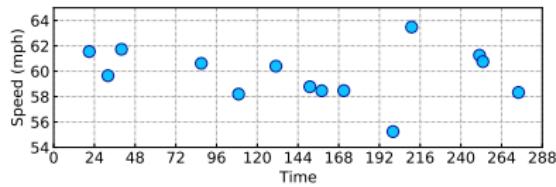


LCR

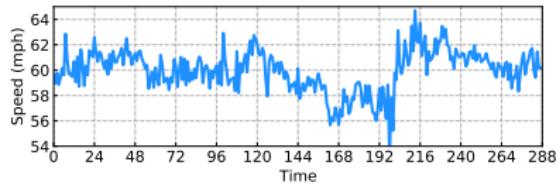
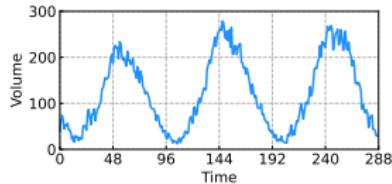
Experiments



↓
Reconstruct
traffic volume?



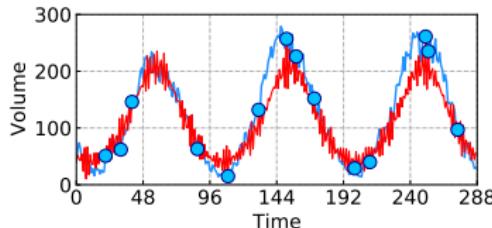
↓
Reconstruct
traffic speed?



- How to utilize the global trends of traffic time series?
- How to produce local consistency of traffic data?

Experiments

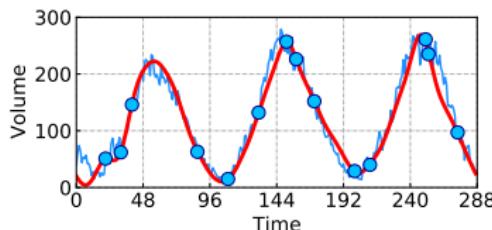
- Substantial performance gains?



CircNNM:

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

↓ Plus **local** time series trends



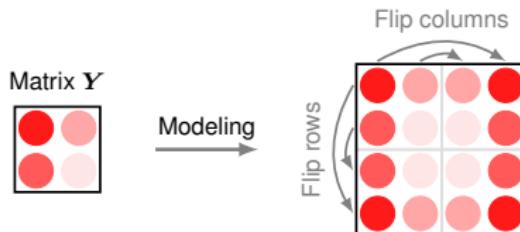
LCR:

$$\begin{aligned} \min_{\boldsymbol{x}} \quad & \|\mathcal{C}(\boldsymbol{x})\|_* + \frac{\gamma}{2} \|\boldsymbol{\ell} * \boldsymbol{x}\|_2^2 \\ \text{s. t. } \quad & \|\mathcal{P}_\Omega(\boldsymbol{x} - \boldsymbol{y})\|_2 \leq \epsilon \end{aligned}$$

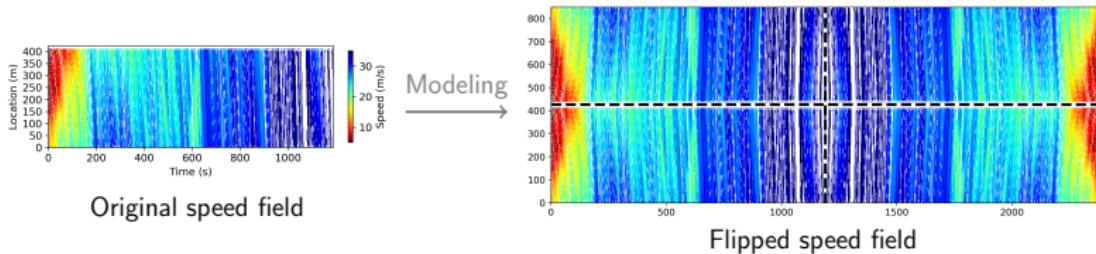
Experiments

Speed field reconstruction³

- Flipping operation on a matrix:



- Flipping operation on a speed field of vehicular traffic flow:



³Highway Drone (HighD) dataset at <https://www.hightd-dataset.com/>

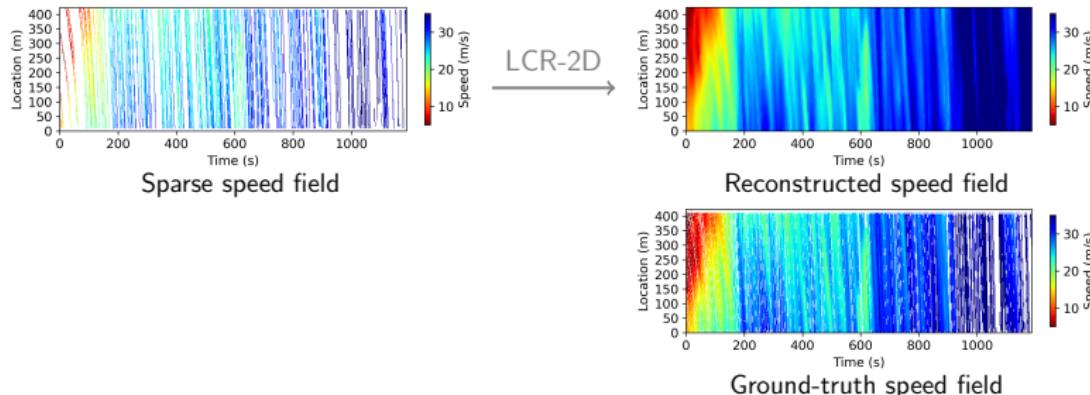
Experiments

Speed field reconstruction⁴

- Scenario: Mask trajectories of 70% vehicles
- LCR-2D on partially observed $\mathbf{Y} \in \mathbb{R}^{N \times T}$:

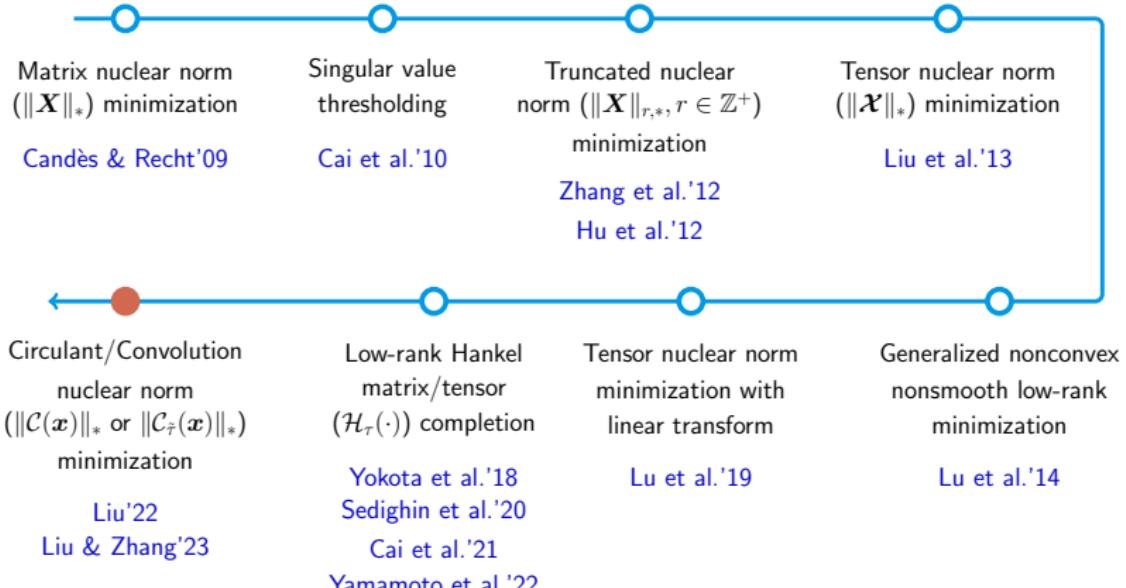
$$\min_{\mathbf{X}} \underbrace{\|\mathcal{C}(\mathbf{X})\|_*}_{\text{global trend}} + \frac{\gamma}{2} \underbrace{\|(\ell_s \ell^\top) * \mathbf{X}\|_F^2}_{\text{local trend}}$$

s.t. $\|\mathcal{P}_\Omega(\mathbf{X} - \mathbf{Y})\|_F \leq \epsilon$



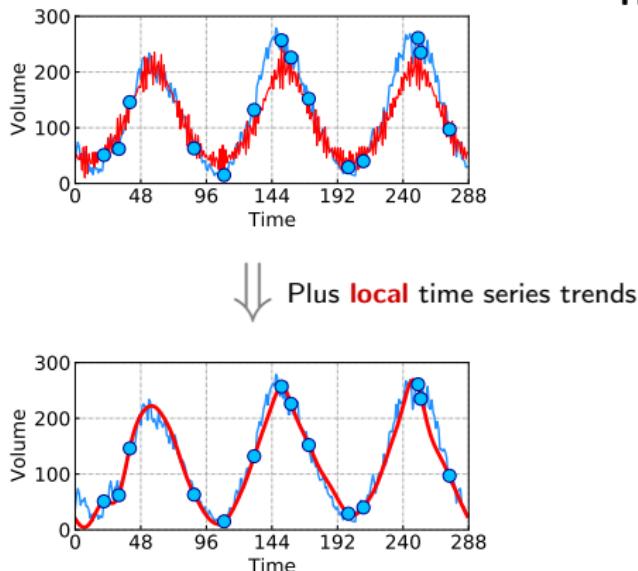
⁴Highway Drone (HighD) dataset at <https://www.hightd-dataset.com/>

Contributions



(Ours) LCR:

- ✓ Local trend modeling
- ✓ An FFT implementation



Highlights:

- Rethinking the importance of local trend modeling in traffic data imputation tasks.
- Finding a unified **global and local trend** modeling framework whose optimization can be efficiently solved by **FFT**:

$$\min_{\mathbf{x}} \underbrace{\|\mathcal{C}(\mathbf{x})\|_*}_{\text{global}} + \frac{\gamma}{2} \underbrace{\|\ell * \mathbf{x}\|_2^2}_{\text{local}}$$

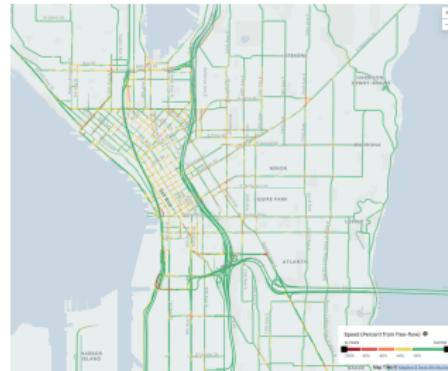
$$\text{s. t. } \|\mathcal{P}_\Omega(\mathbf{x} - \mathbf{y})\|_2 \leq \epsilon$$

Revisit Traffic Prediction

- Uber (hourly) movement speed data⁵



NYC movement



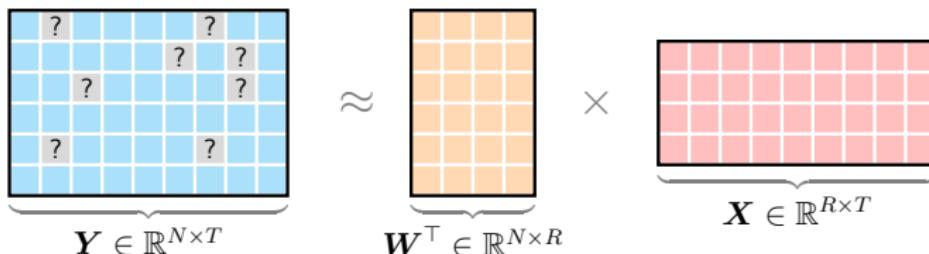
Seattle movement

- {road segment, time step (hour), average speed}
- Computing hourly speed: Road segments have 5+ unique trips.
- Challenge: Forecasting network-wide traffic states with sparse data.

⁵<https://movement.uber.com/> (not available now)

Matrix Factorization

A simple approach to reconstruct missing values.



MF (Koren et al.'09)

Estimating low-dimensional \mathbf{W}, \mathbf{X} :

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

on data \mathbf{Y} w/ observed index set Ω .

- ✓ Learn from sparse data
- ✓ Spatial factor matrix \mathbf{W}
- ✓ Temporal factor matrix \mathbf{X}
- ✗ Temporal correlations?

Temporal Matrix Factorization

Vector autoregression (VAR) on the temporal factor matrix.

$$\begin{array}{c} \text{?} & & \text{?} \\ & \text{?} & \text{?} \\ \text{?} & & \text{?} \\ \text{?} & & \text{?} \end{array} \underbrace{\quad}_{\mathbf{Y} \in \mathbb{R}^{N \times T}} \approx \underbrace{\quad}_{\mathbf{W}^\top \in \mathbb{R}^{N \times R}} \times \underbrace{\quad}_{\mathbf{X} \in \mathbb{R}^{R \times T}}$$

↓ **\mathbf{X} is time series?**

$$\begin{array}{c} \text{?} & & \text{?} \\ & \text{?} & \text{?} \\ \text{?} & & \text{?} \\ \text{?} & & \text{?} \end{array} \underbrace{\quad}_{\mathbf{Y} \in \mathbb{R}^{N \times T}} \approx \underbrace{\quad}_{\mathbf{W}^\top \in \mathbb{R}^{N \times R}} \times \underbrace{\quad}_{\begin{array}{ccccccccc} \mathbf{x}_{t-2} & \mathbf{x}_{t-1} & \mathbf{x}_t & \mathbf{x}_{t+1} & \mathbf{x}_{t+2} \\ \dots & & & & \\ t-3 & t-2 & t-1 & t & t+1 & t+2 & t+3 \end{array} \xrightarrow{\text{time step}}} \dots \Big\} R$$

Why? $\mathbf{X} \in \mathbb{R}^{R \times T}$ is the low-dimensional representation of time series dynamics of $\mathbf{Y} \in \mathbb{R}^{N \times T}$.

Temporal Matrix Factorization

Vector autoregression (VAR) on the temporal factor matrix.

MF (Koren et al.'09)

Estimating low-dimensional \mathbf{W}, \mathbf{X} :

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

on data \mathbf{Y} w/ observed index set Ω .

dth-order VAR

$$+ \quad \mathbf{x}_t = \sum_{k=1}^d \mathbf{A}_k \mathbf{x}_{t-k} + \underbrace{\boldsymbol{\epsilon}_t}_{\mathcal{N}(\mathbf{0}, \mathbf{I})}$$

w/ coefficients $\{\mathbf{A}_k\}$.

↓
Yu et al.'16
Chen & Sun'22

$$\min_{\mathbf{W}, \mathbf{X}, \{\mathbf{A}_k\}_{k=1}^d} \frac{1}{2} \underbrace{\|\mathcal{P}_\Omega(\mathbf{Y} - \mathbf{W}^\top \mathbf{X})\|_F^2}_{\text{MF on data } \mathbf{Y}} + \frac{\gamma}{2} \underbrace{\sum_{t=d+1}^T \left\| \mathbf{x}_t - \sum_{k=1}^d \mathbf{A}_k \mathbf{x}_{t-k} \right\|_2^2}_{\text{VAR on temporal factors } \mathbf{X}}$$

Bayesian Temporal Matrix Factorization

Hankel Temporal Matrix Factorization

Discovering Dynamic Patterns from Spatiotemporal Data with Time-Varying Low-Rank Autoregression

IEEE Transactions on Knowledge and Data Engineering, 2024

<https://doi.org/10.1109/TKDE.2023.3294440>



Xinyu Chen



Chengyuan Zhang*



Xiaoxu Chen



Nicolas Saunier



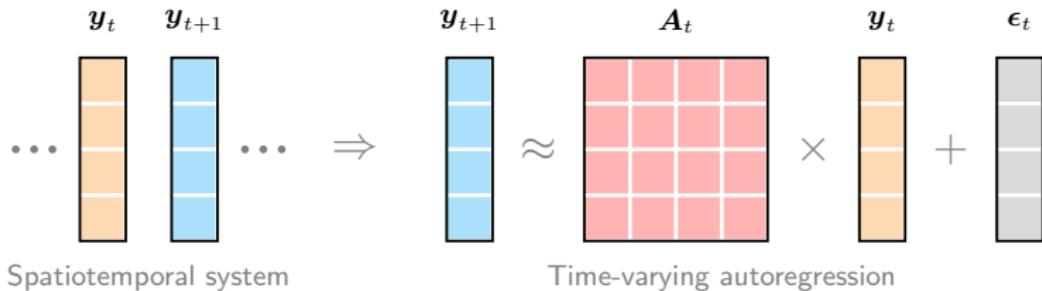
Lijun Sun

Materials:

- GitHub: <https://github.com/xinyuchen/vars>
- Blog: https://spatiotemporal-data.github.io/posts/time_varying_model

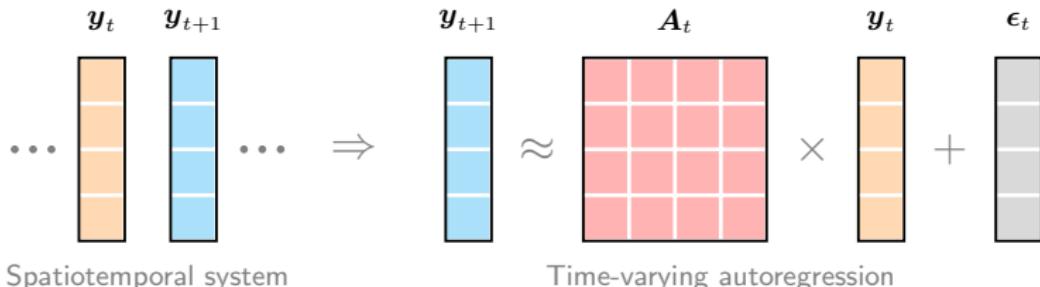
Autoregression

- How to characterize dynamical systems?



Autoregression

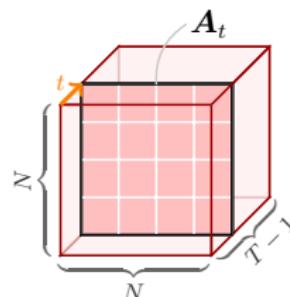
- How to characterize dynamical systems?

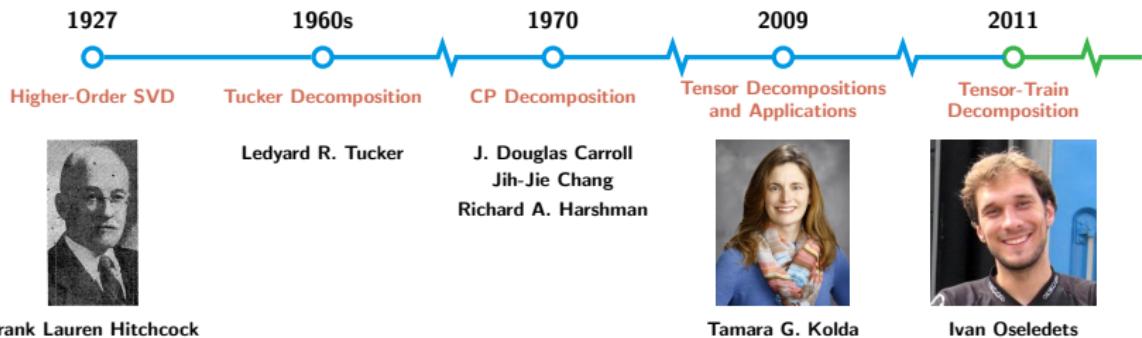


- On spatiotemporal systems $\mathbf{Y} \in \mathbb{R}^{N \times T}$:

$$\underbrace{\mathbf{y}_{t+1} = \mathbf{A}\mathbf{y}_t + \epsilon_t}_{\text{time-invariant (e.g., DMD)}} \quad \text{v.s.} \quad \underbrace{\mathbf{y}_{t+1} = \mathbf{A}_t \mathbf{y}_t + \epsilon_t}_{\text{time-varying}}$$

- How to discover spatial/temporal modes (patterns) from the tensor $\mathcal{A} \triangleq \{\mathbf{A}_t\}_{t \in [T-1]}$?

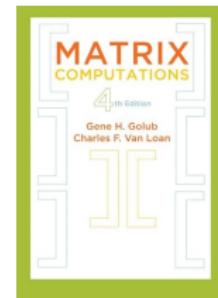




Time-Varying Autoregression

- Tensor factorization⁶:

$$\mathcal{A} = \underbrace{\mathcal{G} \times_1 \mathbf{W} \times_2 \mathbf{V} \times_3 \mathbf{X}}_{\text{Tucker decomposition}}$$
$$\Updownarrow$$
$$\mathbf{A}_t = \mathcal{G} \times_1 \underbrace{\mathbf{W}}_{\text{spatial modes}} \times_2 \mathbf{V} \times_3 \underbrace{\mathbf{x}_t^\top}_{\text{temporal modes}}$$



- (Ours) Time-varying low-rank autoregression:

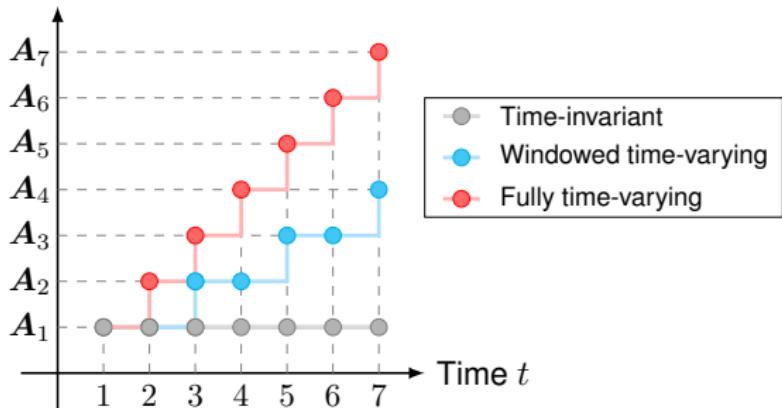
$$\min_{\mathcal{G}, \mathbf{W}, \mathbf{V}, \mathbf{X}} \frac{1}{2} \sum_{t \in [T-1]} \left\| \mathbf{y}_{t+1} - \underbrace{(\mathcal{G} \times_1 \mathbf{W} \times_2 \mathbf{V} \times_3 \mathbf{x}_t^\top)}_{\text{tensor factorization}} \mathbf{y}_t \right\|_2^2$$

⁶ \times_k , $\forall k$ is the mode- k product between tensor and matrix/vector.

- On the data $\mathbf{Y} \in \mathbb{R}^{N \times T}$:

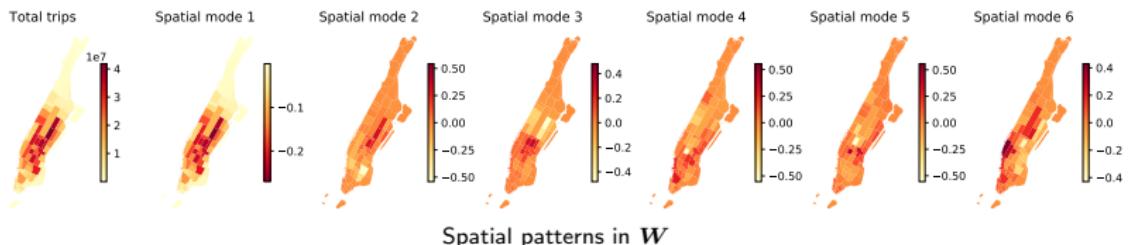
$$\underbrace{\mathbf{y}_{t+1} = \mathbf{A}\mathbf{y}_t + \epsilon_t}_{\text{time-invariant (e.g., DMD)}} \quad \text{v.s.} \quad \underbrace{\mathbf{y}_{t+1} = \mathbf{A}_t \mathbf{y}_t + \epsilon_t}_{\text{fully time-varying (ours)}}$$

Coefficients

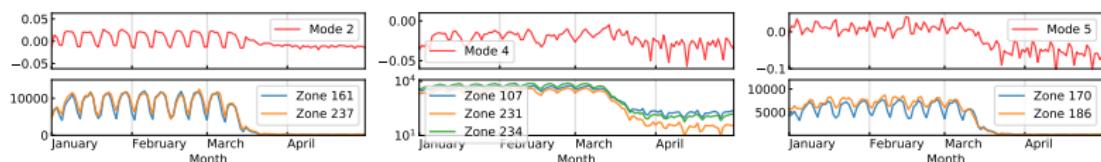


NYC Taxi Data

- NYC taxi dataset (pickup)



Spatial patterns in W



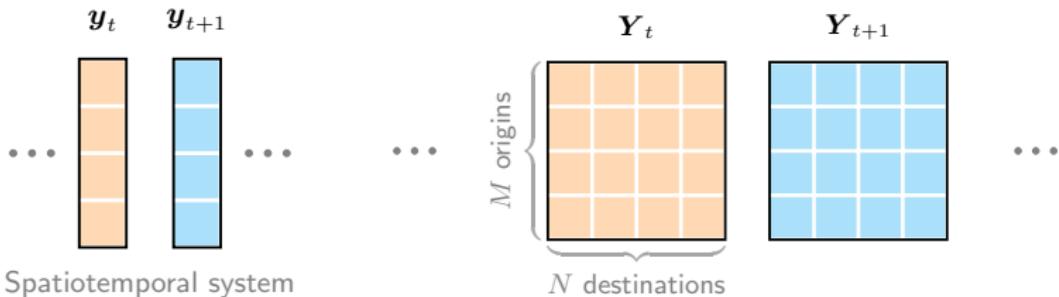
Pattern #2 & taxi trips (2020)

Pattern #4 & taxi trips (2020)

Pattern #5 & taxi trips (2020)

Concluding Remark

- Discovering **spatial/temporal patterns** from 2D and 3D spatiotemporal systems with unsupervised learning:
 - Time-varying autoregression **on the data**
 - Tensor factorization **on the coefficients**





MENS
MANUS AND
MACHINA

Thanks for your attention!

Any Questions?

Slides: <https://xinychen.github.io/slides/informs24.pdf>

About me:

- 🏠 Homepage: <https://xinychen.github.io>
- ✉️ How to reach me: chenxy346@gmail.com
- ✉️ Or send to: xinychen@mit.edu