



**POLYTECHNIQUE
MONTRÉAL**

UNIVERSITÉ
D'INGÉNIERIE



Matrix and Tensor Models for Spatiotemporal Traffic Data Imputation and Forecasting

Ph.D. Defense

Xinyu Chen

Polytechnique Montreal, Canada

December 11, 2023



President
Prof. Francesco Ciari
Polytechnique Montréal



Supervisor
Prof. Nicolas Saunier
Polytechnique Montréal



Co-supervisor
Prof. Lijun Sun
McGill University



Member
Prof. James Goulet
Polytechnique Montréal



External member
Prof. Guillaume Rabusseau
Université de Montréal

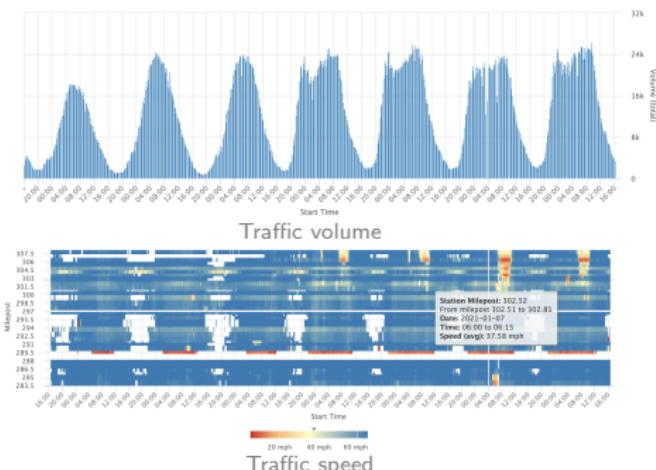
Outline

- **Background**
- **Literature Review**
- **Nonstationary Temporal Matrix Factorization**
- **Low-Rank Autoregressive Tensor Completion**
- **Laplacian Convolutional Representation**
 - Motivation
 - Reformulate Laplacian Regularization
 - Traffic Time Series Imputation
- **Hankel Tensor Factorization**
 - Motivation
 - Hankel Structure
- **Experiments**
- **Conclusion**

Multivariate Traffic Time Series

Many spatiotemporal traffic time series data are in the form of **matrix**.

- Example: Portland highway traffic data¹.



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

¹<https://portal.its.pdx.edu/home>

Multiple Data Behaviors

Spatiotemporal traffic data are time series, but they involve multiple data behaviors.

- Incompleteness & sparsity
- High-dimensionality
- Multidimensionality
- Noises & outliers
- Nonstationarity
-

In addition, spatiotemporal correlations are also very important.

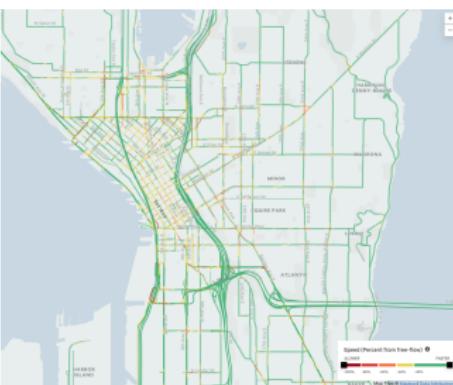
Multiple Data Behaviors

Sparsity & high-dimensionality

- Uber (hourly) movement speed data²



NYC movement



Seattle movement

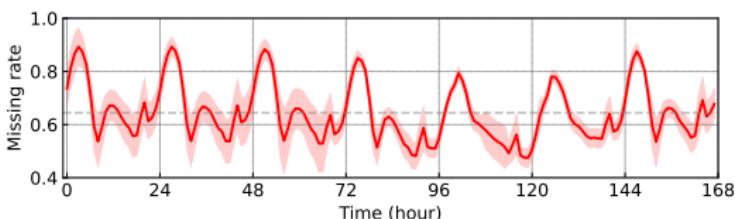
- The average speed on a given road segment for each hour of each day.
- Hourly speeds are computed when road segments have 5+ unique trips.
- **Issue:** insufficient sampling of ridesharing vehicles on the road network.

²<https://movement.uber.com/>

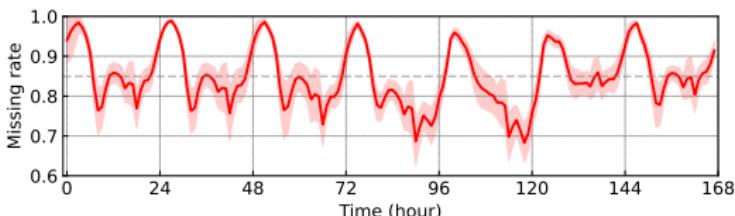
Multiple Data Behaviors

Sparsity & high-dimensionality

- **NYC** movement speed data (2019)
 - 98,210 road segments & 8,760 time steps (hours)
 - Overall missing rate: 64.43%

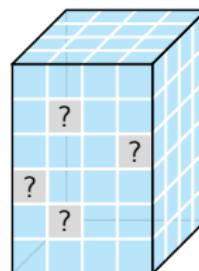
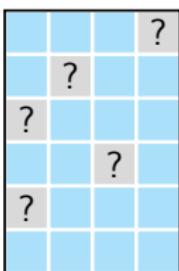


- **Seattle** movement speed data (2019)
 - 63,490 road segments & 8,760 time steps (hours)
 - Overall missing rate: 84.95%



Problem Formulation

- **Objective A:** Given a multivariate time series data like $\mathbf{Y} \in \mathbb{R}^{N \times T}$ or a multidimensional time series data like $\mathcal{Y} \in \mathbb{R}^{M \times N \times T}$ with the observed index set Ω , impute the missing values of the data.

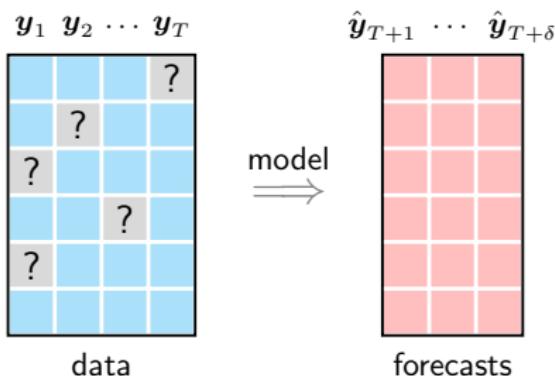


[Q]

- How to reconstruct missing values from observed data?
 - Matrix completion: From $\mathcal{P}_\Omega(\mathbf{Y})$ (observed) to $\mathcal{P}_\Omega^\perp(\mathbf{Y})$ (unobserved)
 - Tensor completion: From $\mathcal{P}_\Omega(\mathcal{Y})$ (observed) to $\mathcal{P}_\Omega^\perp(\mathcal{Y})$ (unobserved)
- How to make use of spatiotemporal correlations?
- How to make use of traffic time series dynamics?

Problem Formulation

- **Objective B:** Given a partially observed data $\mathbf{Y} \in \mathbb{R}^{N \times T}$ consisting of time series $\mathbf{y}_1, \dots, \mathbf{y}_T \in \mathbb{R}^N$, forecast data points $\hat{\mathbf{y}}_{T+\delta}, \delta \in \mathbb{N}^+$.

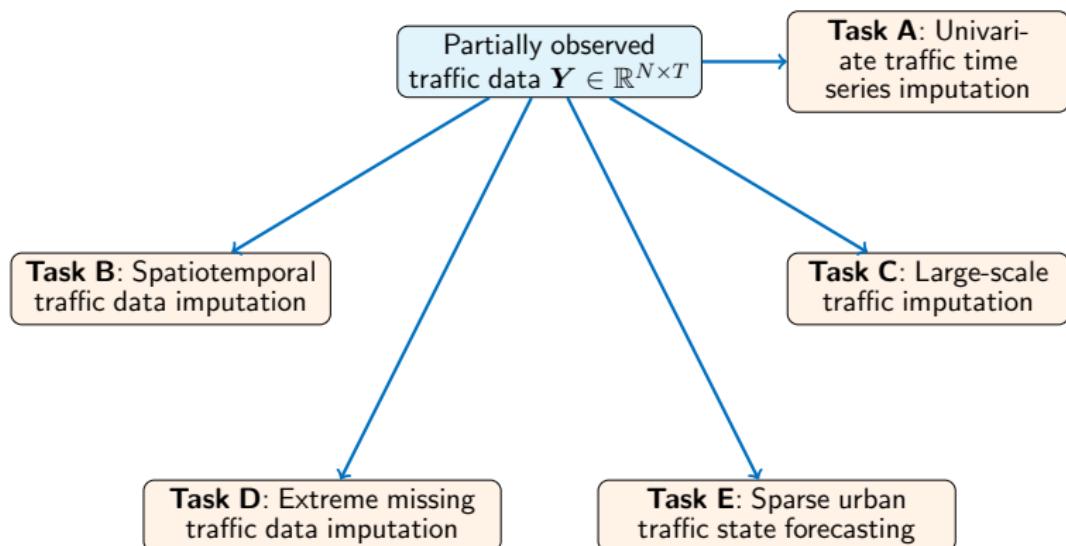


[Q]

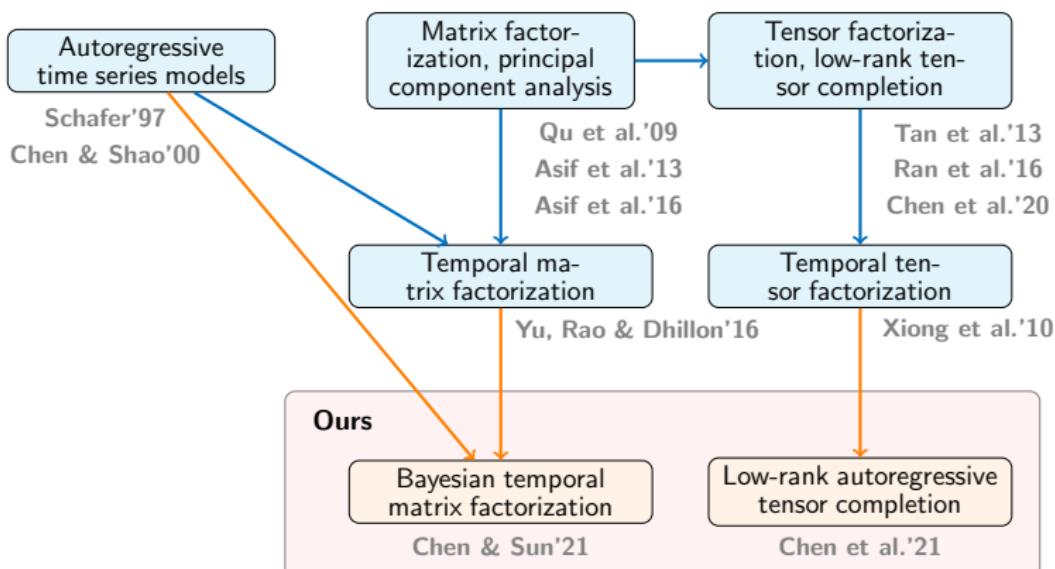
- How to learn from *high-dimensional* and *sparse* data?
- How to model *nonstationarity* in time series?
- How to perform forecasting on these time series?

Whole Picture

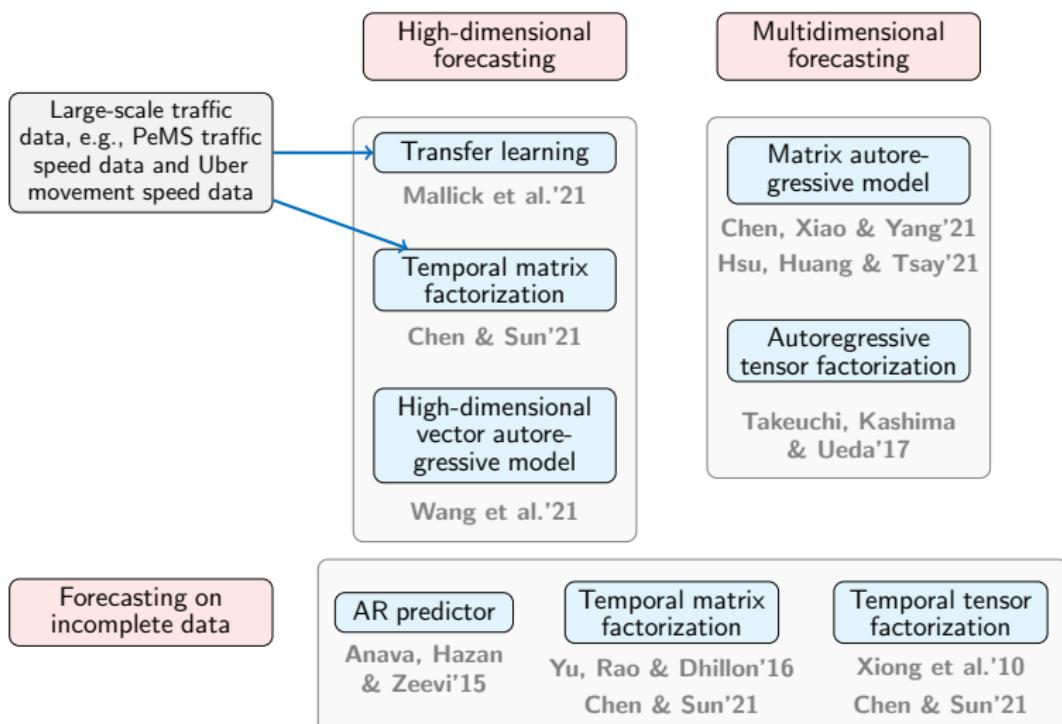
We are working on **spatiotemporal traffic data modeling**.



Spatiotemporal Traffic Data Imputation

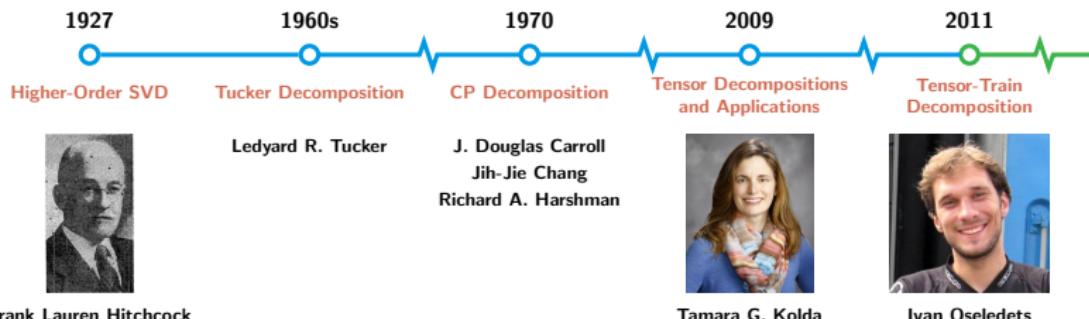


Spatiotemporal Traffic Forecasting

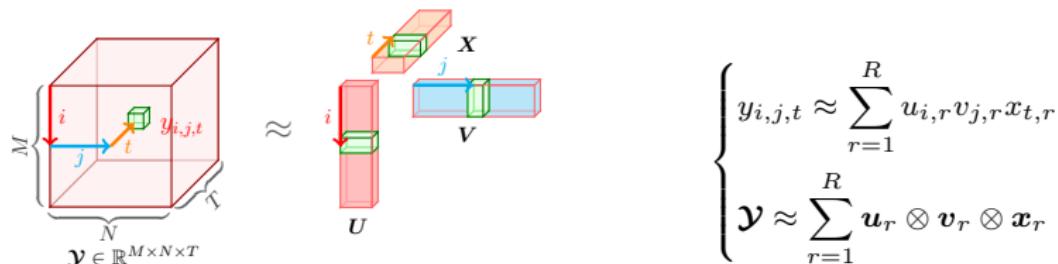


Tensor Factorization

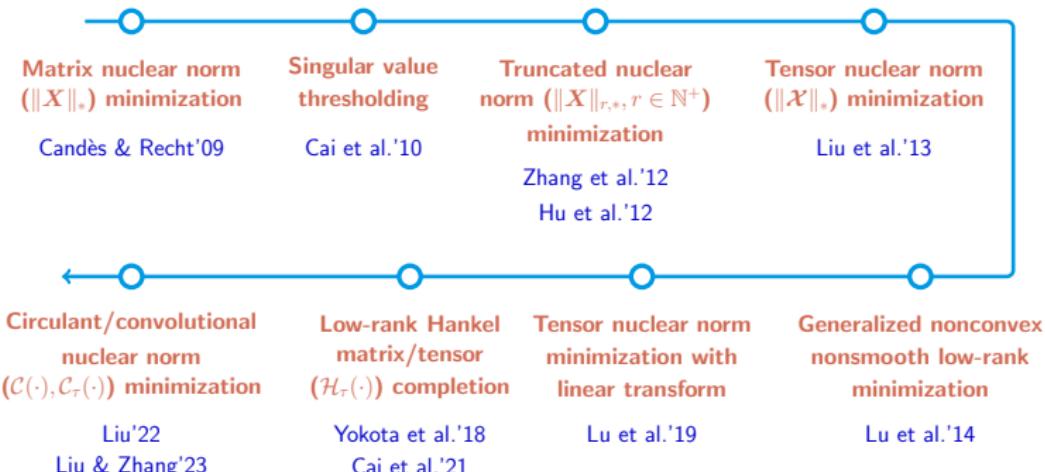
- Revisit tensor factorization



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



Matrix/Tensor Completion

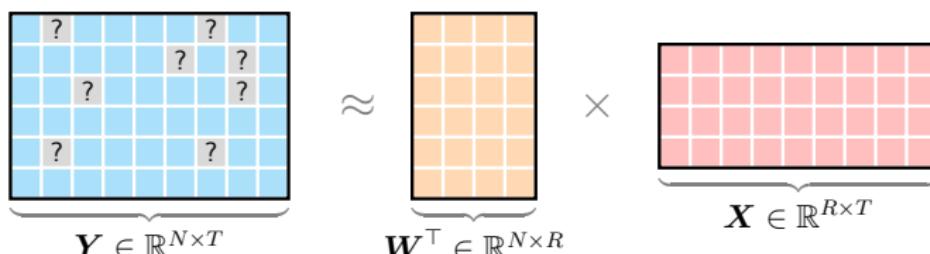


This research

- Integrate temporal modeling techniques (e.g., temporal smoothing and time series autoregression) into low-rank matrix and tensor methods
- Implement spatiotemporal traffic data imputation and forecasting on partially observed data

Matrix Factorization

A simple approach to reconstruct missing values.



MF (Koren et al.'09)

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

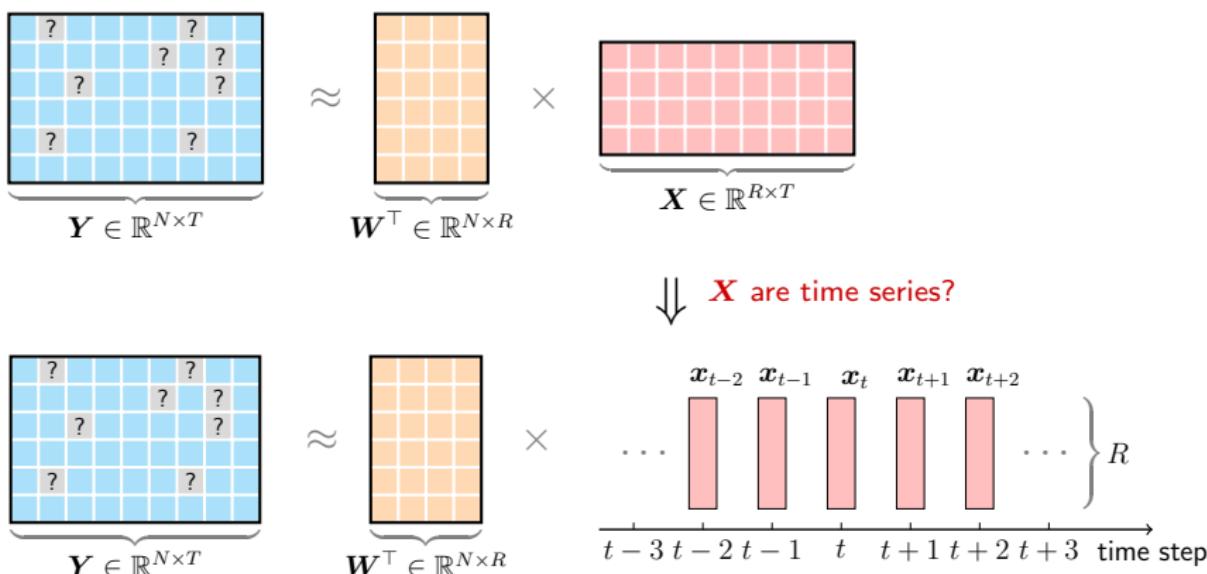
$$\min_{\mathbf{W}, \mathbf{X}} \frac{1}{2} \|\mathcal{P}_\Omega(\mathbf{Y} - \mathbf{W}^\top \mathbf{X})\|_F^2$$

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

- ✓ Learn from sparse data
- ✗ Temporal correlations
- ✗ Time series forecasting

Temporal Matrix Factorization

Vector autoregression (VAR) on the temporal factor matrix.



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

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

$$\mathbf{x}_t = \sum_{k=1}^d \mathbf{A}_k \mathbf{x}_{t-k} + \epsilon_t$$

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

↓ Yu et al.'16
Chen & Sun'21

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

Nonstationary Temporal Matrix Factorization

Nonstationary temporal matrix factorization (NoTMF)

Given any partially observed time series data $\mathbf{Y} \in \mathbb{R}^{N \times T}$ with observed index set Ω , then we assume a season- m differencing on the latent temporal factors:

$$\begin{aligned} \min_{\mathbf{W}, \mathbf{X}, \{\mathbf{A}_k\}_{k=1}^d} & \frac{1}{2} \|\mathcal{P}_\Omega(\mathbf{Y} - \mathbf{W}^\top \mathbf{X})\|_F^2 + \frac{\rho}{2} (\|\mathbf{W}\|_F^2 + \|\mathbf{X}\|_F^2) \\ & + \frac{\gamma}{2} \sum_{t=d+m+1}^T \left\| (\mathbf{x}_t - \mathbf{x}_{t-m}) - \sum_{k=1}^d \mathbf{A}_k (\mathbf{x}_{t-k} - \mathbf{x}_{t-m-k}) \right\|_2^2 \end{aligned}$$

- First-order differencing $\mathbf{x}'_t = \mathbf{x}_t - \mathbf{x}_{t-1}$.
 - Second-order differencing $\mathbf{x}''_t = (\mathbf{x}_t - \mathbf{x}_{t-1}) - (\mathbf{x}_{t-1} - \mathbf{x}_{t-2})$.
 - Twice-differenced series $\mathbf{x}'''_t = (\mathbf{x}_t - \mathbf{x}_{t-m}) - (\mathbf{x}_{t-1} - \mathbf{x}_{t-m-1})$.
- ✓ Stationarizing a time series with differencing can improve the prediction.³

³Stationarity and differencing: <https://otexts.com/fpp2/stationarity.html>

Nonstationary Temporal Matrix Factorization

Rewrite NoTMF

- Optimization problem:

$$\begin{aligned} \min_{\mathbf{W}, \mathbf{X}, \mathbf{A}} \quad & \frac{1}{2} \|\mathcal{P}_\Omega(\mathbf{Y} - \mathbf{W}^\top \mathbf{X})\|_F^2 + \frac{\rho}{2} (\|\mathbf{W}\|_F^2 + \|\mathbf{X}\|_F^2) \\ & + \frac{\gamma}{2} \|\mathbf{X} \Psi_0^\top - \mathbf{A} (\mathbf{I}_d \otimes \mathbf{X}) \Psi^\top\|_F^2 \end{aligned}$$

- Alternating minimization (f be the obj.):

$$\left\{ \begin{array}{ll} \mathbf{W} := \{\mathbf{W} \mid \frac{\partial f}{\partial \mathbf{W}} = \mathbf{0}\} & \text{(least squares)} \\ \mathbf{X} := \{\mathbf{X} \mid \frac{\partial f}{\partial \mathbf{X}} = \mathbf{0}\} & \text{(conjugate gradient)} \\ \mathbf{A} := \{\mathbf{A} \mid \frac{\partial f}{\partial \mathbf{A}} = \mathbf{0}\} & \text{(least squares)} \end{array} \right.$$

Nonstationary Temporal Matrix Factorization

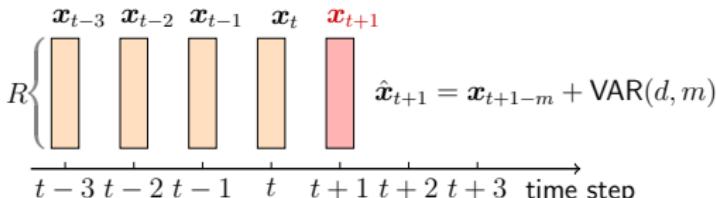
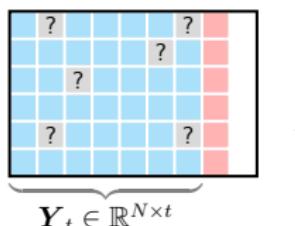
NoTMF forecasting on streaming data?

- NoTMF: Use \mathbf{Y}_t to estimate $\mathbf{W}, \mathbf{X}, \mathbf{A}$.

Implementation

- Estimate $\mathbf{W}, \mathbf{X}, \mathbf{A}$
- Forecast $\hat{\mathbf{x}}_{t+1}$ with VAR
- Return $\hat{\mathbf{y}}_{t+1} = \mathbf{W}^\top \hat{\mathbf{x}}_{t+1}$

- ✓ Sparse input \mathbf{Y}_t
- ✓ Low-dimensional temporal factors
- ✓ Forecast in latent spaces



Nonstationary Temporal Matrix Factorization

NoTMF forecasting on streaming data?

- Online forecasting (Gultekin & Paisley'18): Fix \mathbf{W} and use \mathbf{Y}_{t+1} to update \mathbf{X}, \mathbf{A} .

Implementation

- Estimate \mathbf{X}, \mathbf{A}
 - Forecast $\hat{\mathbf{x}}_{t+2}$ with VAR
 - Return $\hat{\mathbf{y}}_{t+2} = \mathbf{W}^\top \hat{\mathbf{x}}_{t+2}$
- ✓ Sparse input \mathbf{Y}_{t+1}
 - ✓ Fixed spatial factors \mathbf{W}
 - ✓ Forecast in latent spaces

$$\mathbf{Y}_{t+1} \in \mathbb{R}^{N \times (t+1)}$$

A 4x(t+1) matrix \mathbf{Y}_{t+1} with the last column colored red. Some entries in the matrix are marked with question marks.

$$R \begin{cases} \mathbf{x}_{t-3} & \mathbf{x}_{t-2} & \mathbf{x}_{t-1} & \mathbf{x}_t & \mathbf{x}_{t+1} & \mathbf{x}_{t+2} \\ t-3 & t-2 & t-1 & t & t+1 & t+2 & t+3 \end{cases} \text{ time step}$$
$$\hat{\mathbf{x}}_{t+2} = \mathbf{x}_{t+2-m} + \text{VAR}(d, m)$$

A diagram showing a sequence of vectors \mathbf{x}_{t-3} through \mathbf{x}_{t+2} over time steps $t-3$ to $t+3$. The vector \mathbf{x}_{t+2} is highlighted in pink, indicating it is the forecasted value. The formula $\hat{\mathbf{x}}_{t+2} = \mathbf{x}_{t+2-m} + \text{VAR}(d, m)$ is shown below the diagram.

Matrix/Tensor Completion

Cornerstone: Nuclear norm minimization in matrix/tensor completion

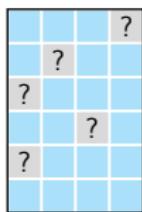
LRMC (Candès & Recht'09)

Estimating the matrix \mathbf{X} :

$$\min_{\mathbf{X}} \|\mathbf{X}\|_*$$

$$\text{s.t. } \mathcal{P}_\Omega(\mathbf{X}) = \mathcal{P}_\Omega(\mathbf{Y})$$

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



$$\mathcal{P}_\Omega(\mathbf{Y}) \in \mathbb{R}^{N \times T}$$

vs.

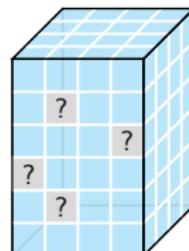
LRTC (Liu et al.'13)

Estimating the tensor \mathcal{X} :

$$\min_{\mathcal{X}} \|\mathcal{X}\|_*$$

$$\text{s.t. } \mathcal{P}_\Omega(\mathcal{X}) = \mathcal{P}_\Omega(\mathcal{Y})$$

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

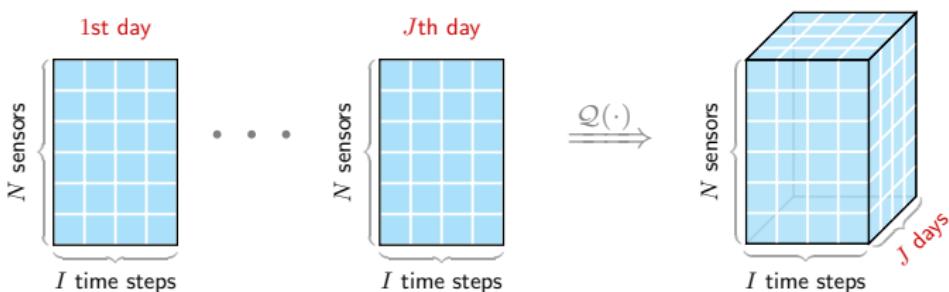


$$\mathcal{P}_\Omega(\mathcal{Y}) \in \mathbb{R}^{N \times I \times J}$$

- **Limitation:** Only cover global consistency

Low-Rank Autoregressive Tensor Completion

- Introduce traffic tensors with day dimension⁴ (Tan et al.'13, Chen et al.'19, ...)



⁴There are $T = IJ$ time steps in total.

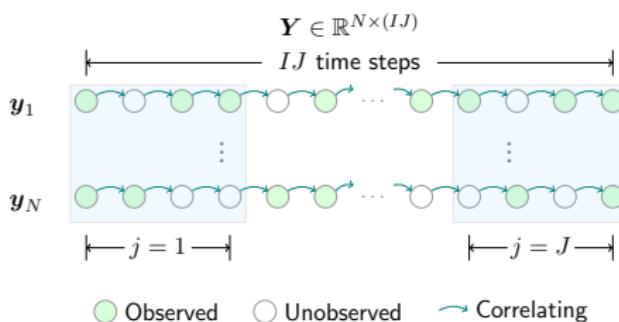
Low-Rank Autoregressive Tensor Completion

- Build temporal correlations with autoregression

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

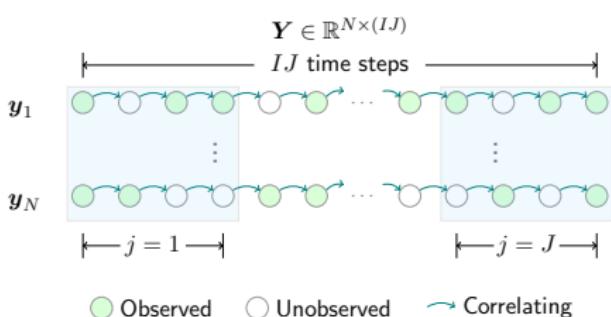
$$\|\mathbf{Y}\|_{\mathbf{A}, \mathcal{H}} \triangleq \sum_{n,t} \left(y_{n,t} - \sum_k \mathbf{a}_{n,k} y_{n,t-h_k} \right)^2$$

with the time lag set $\mathcal{H} = \{h_1, \dots, h_d\}$ and the coefficient matrix $\mathbf{A} \in \mathbb{R}^{N \times d}$.

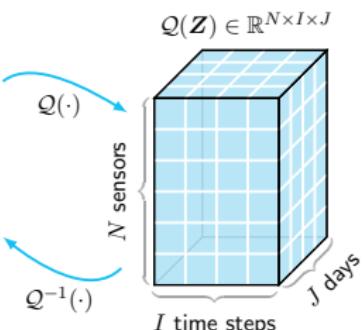


Low-Rank Autoregressive Tensor Completion

Local consistency w/ autoregression



Global consistency w/ tensor structure



LATC

Optimization problem:

$$\begin{aligned} \min_{\mathbf{Z}, \mathbf{A}} \quad & \|\mathcal{Q}(\mathbf{Z})\|_{r,*} + \frac{\gamma}{2} \|\mathbf{Z}\|_{\mathbf{A}, \mathcal{H}} \\ \text{s.t. } & \mathcal{P}_{\Omega}(\mathbf{Z}) = \mathcal{P}_{\Omega}(\mathbf{Y}) \end{aligned}$$

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

Two subproblems

$$\Rightarrow \begin{cases} \mathbf{Z} := \underset{\mathcal{P}_{\Omega}(\mathbf{Z}) = \mathcal{P}_{\Omega}(\mathbf{Y})}{\arg \min} \|\mathcal{Q}(\mathbf{Z})\|_{r,*} + \frac{\gamma}{2} \|\mathbf{Z}\|_{\mathbf{A}, \mathcal{H}} \\ \mathbf{A} := \frac{1}{2} \|\mathbf{Z}\|_{\mathbf{A}, \mathcal{H}} \end{cases} \quad (\text{Least squares})$$

Low-Rank Autoregressive Tensor Completion

Z -subproblem:

$$\mathbf{Z} := \arg \min_{\mathcal{P}_{\Omega}(\mathbf{Z}) = \mathcal{P}_{\Omega}(\mathbf{Y})} \|\mathcal{Q}(\mathbf{Z})\|_{r,*} + \frac{\gamma}{2} \|\mathbf{Z}\|_{A,\mathcal{H}}$$

- Augmented Lagrangian function:⁵

$$\mathcal{L}(\mathbf{X}, \mathbf{Z}, \mathbf{W}) = \|\mathbf{X}\|_{r,*} + \frac{\gamma}{2} \|\mathbf{Z}\|_{A,\mathcal{H}} + \frac{\lambda}{2} \|\mathbf{X} - \mathcal{Q}(\mathbf{Z})\|_F^2 + \langle \mathbf{W}, \mathbf{X} - \mathcal{Q}(\mathbf{Z}) \rangle + \pi(\mathbf{Z})$$

Implementation

Repeat

- Compute \mathbf{Z}
- Compute \mathbf{A}



Implementation

Repeat

- Repeat
 - # Alternating Direction Method of Multipliers (ADMM)

- Compute \mathbf{X}
- Compute \mathbf{Z}
- Compute \mathbf{W}

- Compute \mathbf{A}

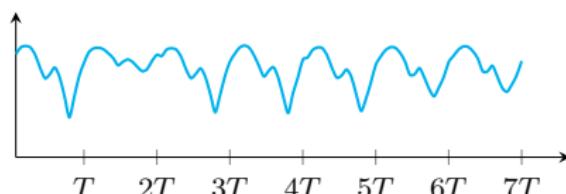
⁵ $\mathbf{W} \in \mathbb{R}^{N \times I \times J}$ (Lagrange multiplier); The indicator function:

$$\pi(\mathbf{Z}) = \begin{cases} 0, & \text{if } \mathcal{P}_{\Omega}(\mathbf{Z}) = \mathcal{P}_{\Omega}(\mathbf{Y}), \\ +\infty, & \text{otherwise.} \end{cases}$$

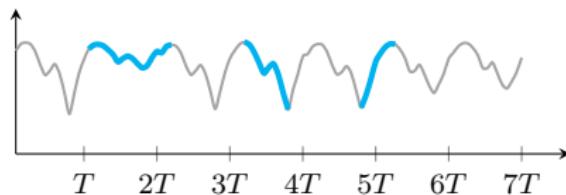
Laplacian Convolutional Representation

Motivation: Time series imputation

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



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



- [Question] How to characterize both global and local trends in sparse time series data?

Laplacian Convolutional Representation

Local trend modeling: Reformulate temporal regularization with circular convolution.

- Intuition of (circulant) Laplacian matrix.

The diagram illustrates the construction of a (Circulant) Laplacian matrix. On the left, five nodes labeled x_1, x_2, x_3, x_4, x_5 are arranged in a circle, connected by curved edges forming a cycle. This is labeled "Undirected and circulant graph". An arrow labeled "Modeling" points to the right, where the matrix \mathbf{L} is given as:

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

(Circulant) Laplacian matrix

- Define Laplacian kernel:

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

\Downarrow

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

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

- (Laplacian) Temporal regularization:

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

Laplacian Convolutional Representation

Global trend modeling
Circulant matrix definition
Literature review of circulant/convolution matrix

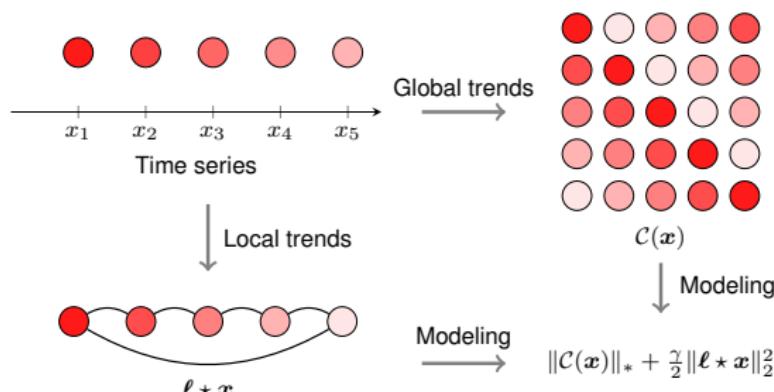
Laplacian Convolutional Representation

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.,

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

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



Laplacian Convolutional Representation

- 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:

$$\begin{cases} \mathbf{x} := \arg \min_{\mathbf{x}} \mathcal{L}(\mathbf{x}, \mathbf{z}, \mathbf{w}) & \text{(Nuclear norm minimization)} \\ \mathbf{z} := \arg \min_{\mathbf{z}} \mathcal{L}(\mathbf{x}, \mathbf{z}, \mathbf{w}) & \text{(Closed-form solution)} \\ = \frac{1}{\lambda + \eta} \mathcal{P}_\Omega(\lambda \mathbf{x} + \mathbf{w} + \eta \mathbf{y}) + \frac{1}{\lambda} \mathcal{P}_\Omega^\perp(\lambda \mathbf{x} + \mathbf{w}) \\ \mathbf{w} := \mathbf{w} + \lambda(\mathbf{x} - \mathbf{z}) & \text{(Standard update)} \end{cases}$$

- Optimize \mathbf{x} ?

$$\|\mathcal{C}(\mathbf{x})\|_* = \|\mathcal{F}(\mathbf{x})\|_1 \quad \& \quad \frac{1}{2} \|\ell * \mathbf{x}\|_2^2 = \frac{1}{2T} \|\mathcal{F}(\ell) \circ \mathcal{F}(\mathbf{x})\|_2^2$$

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

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:

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

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

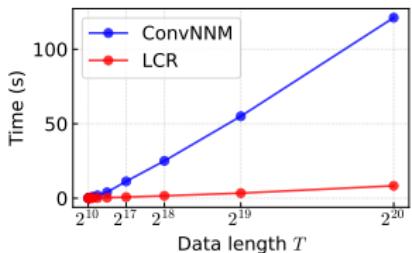
$$\hat{x}_t := \frac{\hat{h}_t}{|\hat{h}_t|} \cdot \max\{0, |\hat{h}_t| - 1/\omega\}, t = 1, \dots, T.$$

Laplacian Convolutional Representation

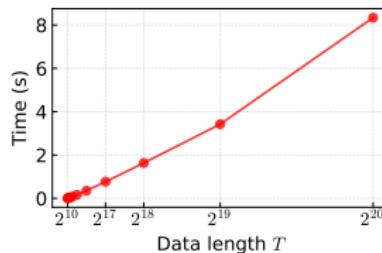
Empirical time complexity

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**⁶ ([Liu'22](#), [Liu & Zhang'23](#))
 - Convolution matrix $C_{\tilde{\tau}}(\mathbf{y}) \in \mathbb{R}^{T \times \tilde{\tau}}$ with kernel size $\tilde{\tau} \in \mathbb{N}^+$
 - Singular value thresholding in $\mathcal{O}(\tilde{\tau}^2 T)$



ConvNNM vs. LCR



LCR

⁶Convolution nuclear norm minimization.

Laplacian Convolutional Representation

Two-Dimensional 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,

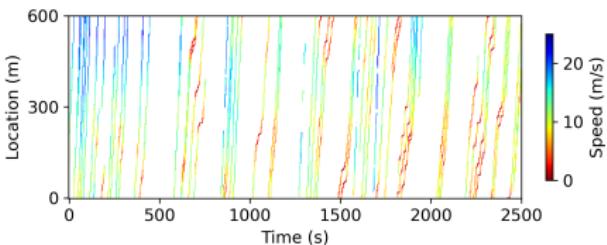
$$\begin{aligned} \min_{\mathbf{X}} \quad & \|\mathcal{C}(\mathbf{X})\|_* + \frac{\gamma}{2} \|(\boldsymbol{\ell}_s \boldsymbol{\ell}^\top) \star \mathbf{X}\|_F^2 \\ \text{s.t. } & \|\mathcal{P}_\Omega(\mathbf{X} - \mathbf{Y})\|_F \leq \epsilon \end{aligned}$$

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

Hankel Tensor Factorization

Motivation: Spatiotemporal data reconstruction

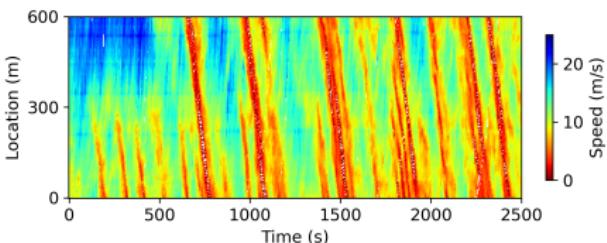
- Speed field reconstruction problem in vehicular traffic flow.



200-by-500 matrix
(NGSIM)



Reconstruct speed field from
5% sparse trajectories?



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

Hankel Tensor Factorization

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

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



Hankel matrix (Source: Twitter)

Hankel Tensor Factorization

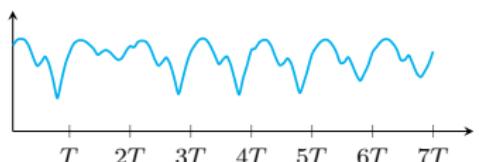
- Hankel matrix

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

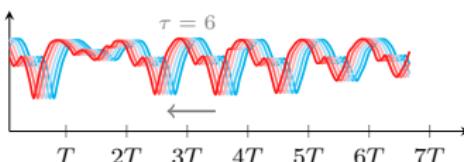
$$\mathcal{H}_\tau(\mathbf{y}) = \begin{bmatrix} y_1 & y_2 \\ y_2 & y_3 \\ y_3 & y_4 \\ y_4 & y_5 \end{bmatrix} \approx \begin{bmatrix} v_1 \\ v_2 \\ v_3 \\ v_4 \end{bmatrix} \otimes \begin{bmatrix} x_1 \\ x_2 \end{bmatrix}$$

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

- Automatic temporal modeling.



Traffic time series



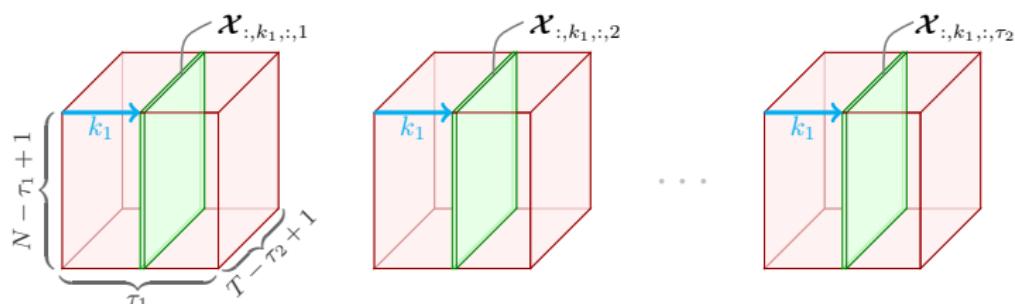
Hankel matrix

Hankel Tensor Factorization

- Hankel tensor: Given any matrix $\mathbf{X} \in \mathbb{R}^{N \times T}$, we have

$$\mathcal{X} \triangleq \mathcal{H}_{\tau_1, \tau_2}(\mathbf{X})$$

- Window lengths: $\tau_1, \tau_2 \in \mathbb{N}^+$;
- Tensor size: $(N - \tau_1 + 1) \times \tau_1 \times (T - \tau_2 + 1) \times \tau_2$;



(Figure) 4th order Hankel tensor: A sequence of third-order tensors.

- Slice: $\mathcal{X}_{:,k_1,:,k_2}$, $\forall k_1, k_2$;
- Slice size: $(N - \tau_1 + 1) \times (T - \tau_2 + 1)$.

Hankel Tensor Factorization

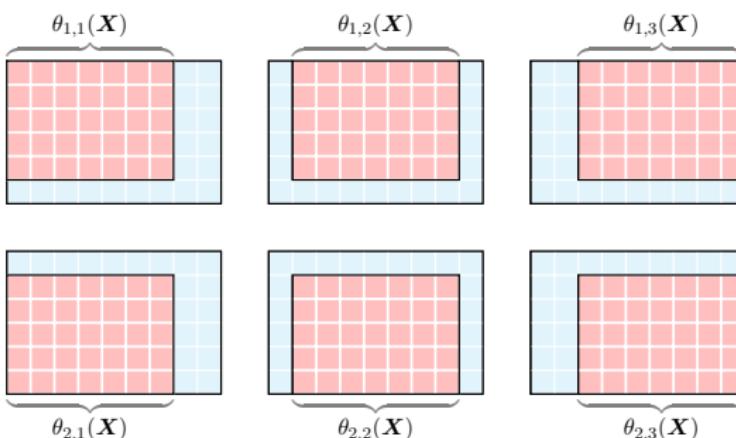
Hankel indexing:

- Sampling function for the Hankelization:

$$\theta_{k_1, k_2}(\mathbf{X}) \triangleq [\mathcal{H}_{\tau_1, \tau_2}(\mathbf{X})]_{:, k_1, :, k_2},$$

referring to the tensor slice with $k_1 \in \{1, \dots, \tau_1\}$, $k_2 \in \{1, \dots, \tau_2\}$.

- [Importance] Developing memory-efficient algorithms.



- Tensor slices $\theta_{k_1, k_2}(\mathbf{X})$ vs. data matrix \mathbf{X}

Hankel Tensor Factorization

Ours:

- Convolutional tensor decomposition (circular convolution \star_{row}):

$$\theta_{k_1, k_2}(\mathbf{Y}) \approx (\mathbf{Q} \star_{\text{row}} \mathbf{s}_{k_1}^{\top})(\mathbf{U} \star_{\text{row}} \mathbf{v}_{k_2}^{\top})^{\top}$$

Baselines:

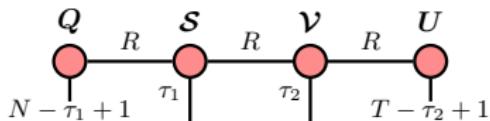
- CP tensor decomposition (Khatri-Rao product \odot):

$$\theta_{k_1, k_2}(\mathbf{Y}) \approx (\mathbf{Q} \odot \mathbf{s}_{k_1}^{\top})(\mathbf{U} \odot \mathbf{v}_{k_2}^{\top})^{\top}$$

- Tensor-train decomposition:

$$\theta_{k_1, k_2}(\mathbf{Y}) \approx (\mathbf{Q} \mathbf{S}_{k_1})(\mathbf{U} \mathbf{V}_{k_2})^{\top}$$

- $\{\mathbf{S}_{k_1}, \mathbf{V}_{k_2}\}$ are **circulant matrices** \Rightarrow convolutional decomposition
- $\{\mathbf{S}_{k_1}, \mathbf{V}_{k_2}\}$ are **diagonal matrices** \Rightarrow CP decomposition



Hankel Tensor Factorization

HTF (convolutional decomposition)

- Optimization problem:

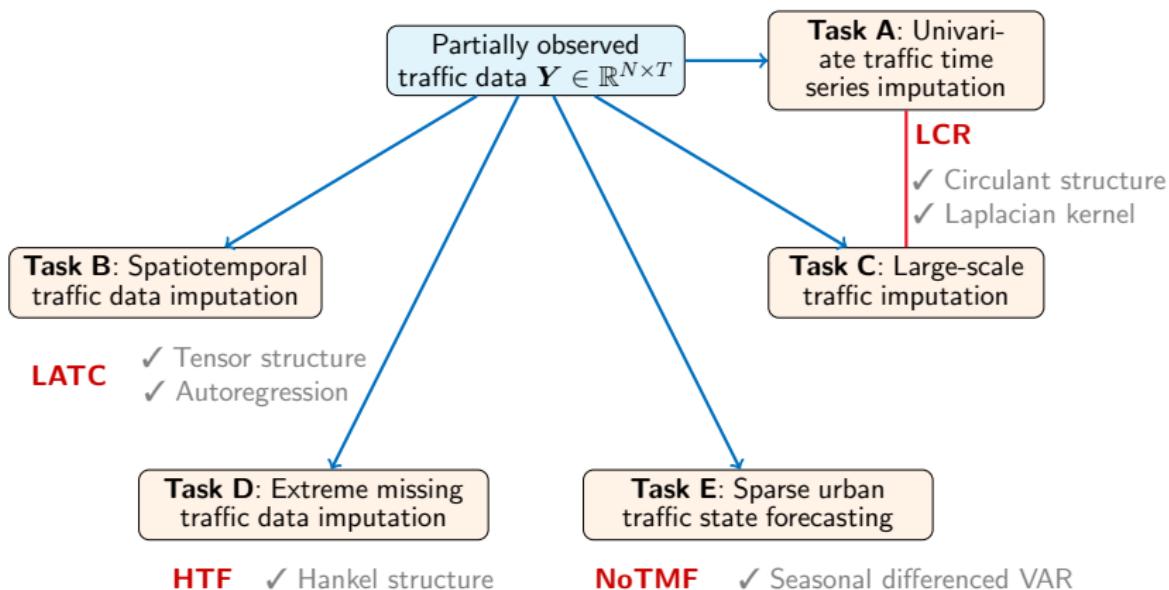
$$\begin{aligned} \min_{\mathbf{Q}, \mathbf{S}, \mathbf{U}, \mathbf{V}} \quad & \frac{1}{2} \sum_{k_1, k_2} \left\| \mathcal{P}_{\Omega_{k_1, k_2}} (\theta_{k_1, k_2}(\mathbf{Y}) - (\mathbf{Q} \star_{\text{row}} \mathbf{s}_{k_1})(\mathbf{U} \star_{\text{row}} \mathbf{v}_{k_2})^\top) \right\|_F^2 \\ & + \frac{\rho}{2} (\|\mathbf{Q}\|_F^2 + \|\mathbf{S}\|_F^2 + \|\mathbf{U}\|_F^2 + \|\mathbf{V}\|_F^2) \end{aligned}$$

- Alternating minimization:

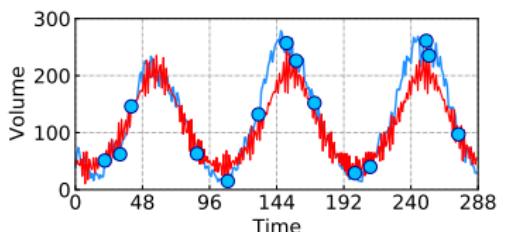
$$\left\{ \begin{array}{ll} \mathbf{Q} := \{\mathbf{Q} \mid \frac{\partial f}{\partial \mathbf{Q}} = \mathbf{0}\} & \text{(conjugate gradient)} \\ \mathbf{s}_{k_1} := \{\mathbf{s}_{k_1} \mid \frac{\partial f}{\partial \mathbf{s}_{k_1}} = \mathbf{0}\}, \forall k_1 & \text{(conjugate gradient)} \\ \mathbf{U} := \{\mathbf{U} \mid \frac{\partial f}{\partial \mathbf{U}} = \mathbf{0}\} & \text{(conjugate gradient)} \\ \mathbf{v}_{k_2} := \{\mathbf{v}_{k_2} \mid \frac{\partial f}{\partial \mathbf{v}_{k_2}} = \mathbf{0}\}, \forall k_2 & \text{(conjugate gradient)} \end{array} \right.$$

Whole Picture

We are working on **spatiotemporal traffic data modeling**.



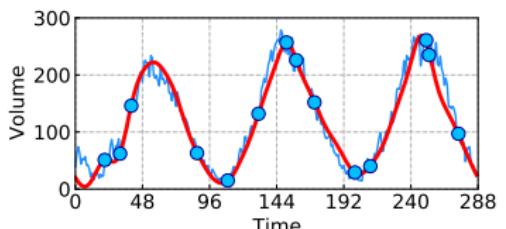
Univariate Traffic Time Series Imputation



CircNNM:

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

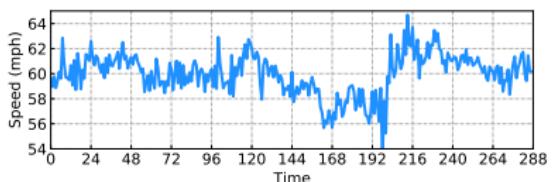
↓ Plus temporal regularization (TR)



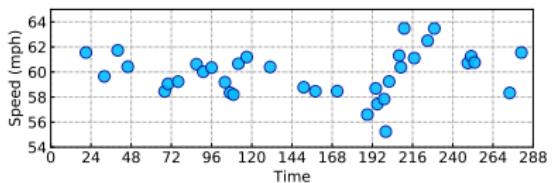
LCR:

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

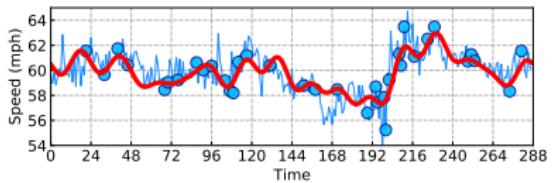
Univariate Traffic Time Series Imputation



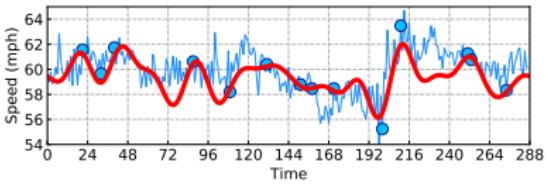
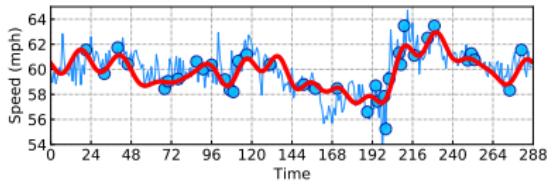
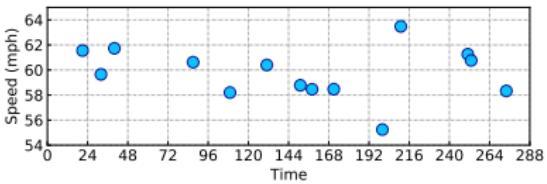
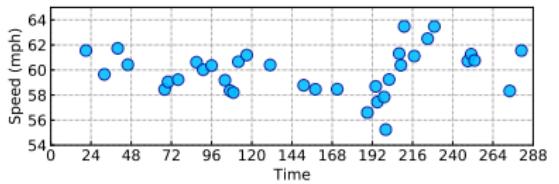
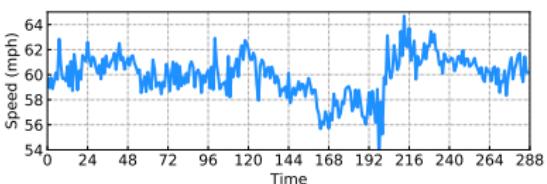
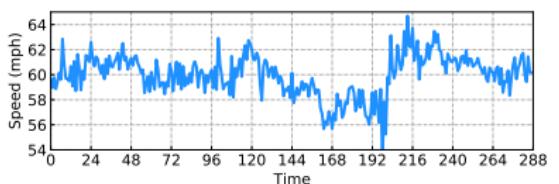
↓ Mask 90% observations



↓ Reconstruct time series



Univariate Traffic Time Series Imputation



Spatiotemporal Traffic Data Imputation

LATC vs. baseline models (in MAPE/RMSE)

- On the Seattle freeway traffic speed dataset ($\mathbf{Y} \in \mathbb{R}^{323 \times 8064}$)

Missing rate	LATC	LAMC	LRTC-TNN	BTMF	SPC
30%, Random Missing	4.90/3.16	5.98/3.73	4.99/3.20	5.91/3.72	5.92/3.62
70%, Random Missing	5.96/3.71	8.02/4.70	6.10/3.77	6.47/3.98	7.38/4.30
90%, Random Missing	7.46/4.50	10.56/5.91	8.08/4.80	8.17/4.81	9.75/5.31
30%, Nonrandom Missing	7.10/4.33	6.99/4.25	6.85/4.21	9.26/5.36	8.87/4.99
70%, Nonrandom Missing	9.40/5.40	9.75/5.60	9.23/5.35	10.47/6.15	11.32/5.92
30%, Block-out Missing	9.43/5.36	27.05/13.66	9.52/5.41	14.33/13.60	11.30/5.84

- On the Portland highway traffic volume dataset ($\mathbf{Y} \in \mathbb{R}^{1156 \times 2976}$)

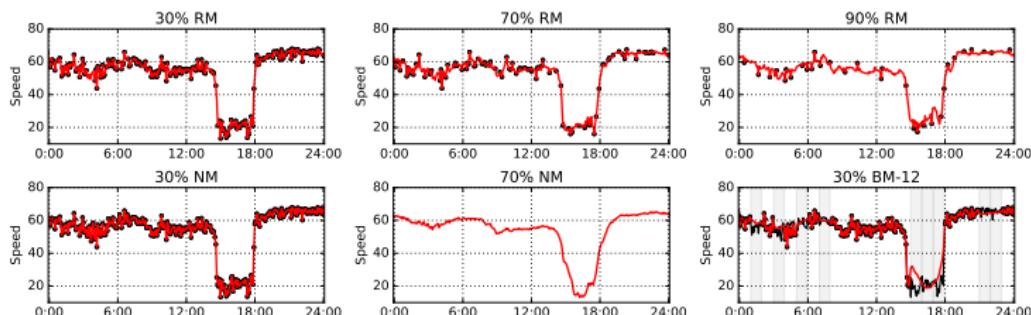
Missing rate	LATC	LAMC	LRTC-TNN	BTMF	SPC
30%, Random Missing	16.95/15.99	17.93/16.03	17.27/16.08	18.22/19.14	21.29/56.73
70%, Random Missing	19.59/18.70	21.26/19.37	19.99/18.73	19.96/22.21	24.35/43.32
90%, Random Missing	23.15/22.83	25.64/23.75	22.90/22.68	23.90/25.71	28.45/39.65
30%, Nonrandom Missing	19.48/19.14	19.93/19.69	19.59/ 18.91	19.55/20.38	26.96/60.33
70%, Nonrandom Missing	27.67/45.03	25.75/28.25	30.26/60.85	23.86/26.74	33.42/47.34
30%, Block-out Missing	24.01/23.50	29.21/27.60	31.74/74.42	27.85/25.68	31.01/60.33

- LATC vs. LAMC: The significance of tensor representation
- LATC vs. LRTC-TNN: The significance of temporal autoregression

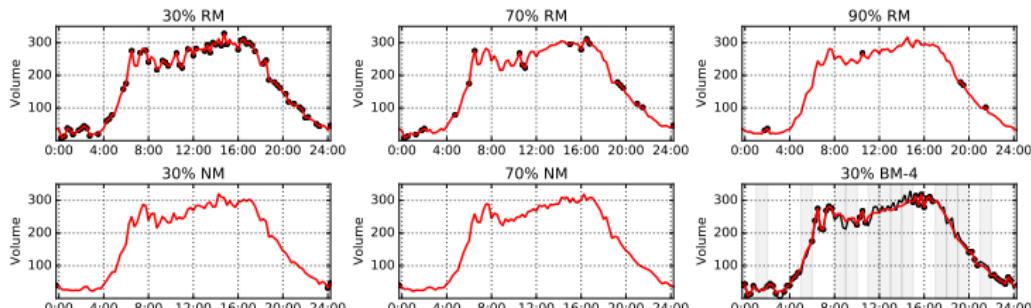
Spatiotemporal Traffic Data Imputation

LATC imputation

- Seattle freeway traffic speed data



- Portland highway traffic volume data



Large-Scale Traffic Data Imputation

LCR vs. baseline models (in MAPE/RMSE)

- PeMS-4W: California freeway traffic speed dataset ($Y \in \mathbb{R}^{11160 \times 8064}$)

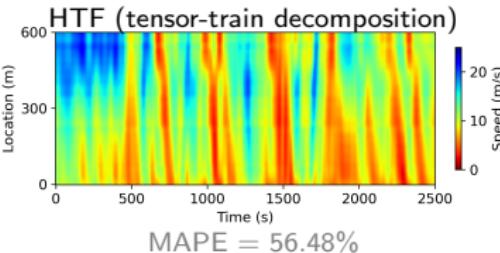
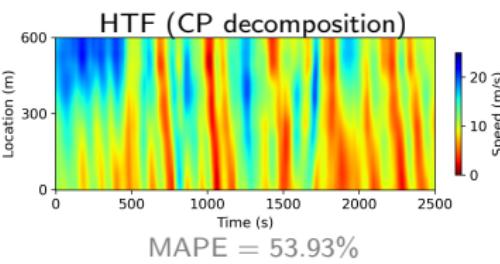
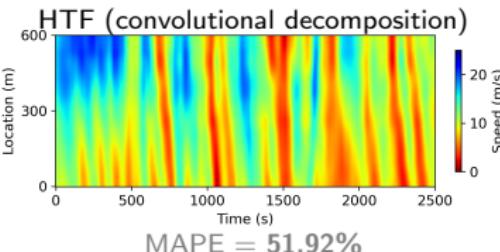
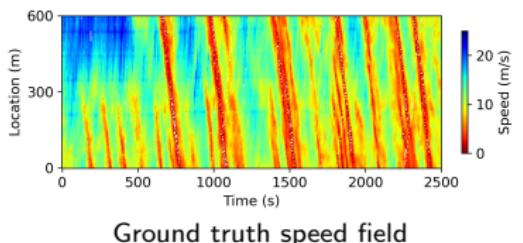
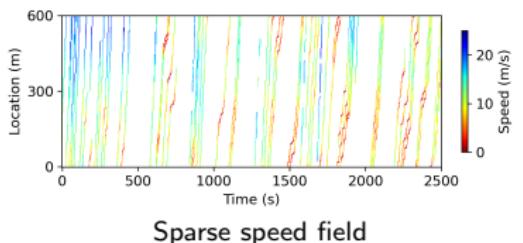
Model	Missing rate			
	30%	50%	70%	90%
LCR-2D	1.50/1.49	1.76/1.69	2.07/2.06	3.19/3.05
LCR_N	1.48/1.50	1.73/1.73	2.07/2.12	3.24/3.22
LCR	1.50/1.49	1.76/1.69	2.08/2.07	3.21/3.06
CTNNM	2.26/1.84	2.67/2.14	3.40/2.66	5.22/3.90
CircNNM	2.26/1.84	2.69/2.15	3.43/2.67	5.34/3.96
LRMC	2.04/1.80	2.43/2.12	3.08/2.66	6.05/4.43
HaLRTC	1.98/1.73	2.22/1.98	2.84/2.49	4.39/3.66
LRTC-TNN	1.68/1.55	1.93/1.77	2.33/2.14	3.40/3.10
NoTMF	2.95/2.65	3.05/2.73	3.33/2.97	5.22/4.71

Results

- LCR-2D > CTNNM: The importance of temporal regularization
- CTNNM \geq CircNNM: Cyclic tensor is superior to circulant matrix
- LCR > LRMC/LRTC: The importance of global/local modeling

$\mathcal{O}(NT \log(NT))$ (FFT) vs. $\mathcal{O}(\min\{N^2T, NT^2\})$ (SVD)

Extreme Missing Traffic Data Imputation



Sparse Urban Traffic State Forecasting

NoTMF

Conclusion

- Data (large-scale, high-dimensional, city-wide, sparse)
- Modeling (meaningfulness and importance of temporal correlations)

Conclusion

Low-Rank Autoregressive Tensor Completion (LATC):

- (Highlight) Global & local consistency
 - ✓ Tensor structure $\|\mathcal{Q}(\mathbf{Z})\|_{r,*}$
 - ✓ Autoregression $\|\mathbf{Z}\|_{\mathbf{A}, \mathcal{H}}$

Laplacian Convolutional Representation (LCR):

- (Solution) Global & local time series trends
 - Global trend modeling: $\|\mathcal{C}(\mathbf{x})\|_*$
 - Local trend modeling: $\|\boldsymbol{\ell} \star \mathbf{x}\|_2^2$
- (Highlight) A unified framework with the **FFT** implementation.

Hankel Tensor Factorization (HTF):

- (Highlight) Memory-efficient **Hankel indexing** & convolutional parameterization.

Collaborators



Dr. HanQin Cai



Xiaoxu Chen



Dr. Zhanhong Cheng



Chengyuan Zhang



Dr. Xi-Le Zhao

References

A short list:

- [Candès & Recht'09] Exact matrix completion via convex optimization.
Foundations of Computational Mathematics, 9 (6), 717-772.
- [Cai et al.'10] A singular value thresholding algorithm for matrix completion
- [Zhang et al.'12] Matrix completion by truncated nuclear norm regularization
- [Hu et al.'12] Fast and accurate matrix completion via truncated nuclear norm regularization
- [Lu et al.'14] Generalized Nonconvex Nonsmooth Low-Rank Minimization
- [Lu et al.'19] Tensor Robust Principal Component Analysis with A New Tensor Nuclear Norm
- [Yokota et al.'18] Missing Slice Recovery for Tensors Using a Low-rank Model in Embedded Space
- [Cai et al.'21] Accelerated Structured Alternating Projections for Robust Spectrally Sparse Signal Recovery
- [Liu'22]
- [Liu & Zhang'23]



POLYTECHNIQUE
MONTRÉAL

UNIVERSITÉ
D'INGÉNIERIE



Thanks for your attention!

Any Questions?

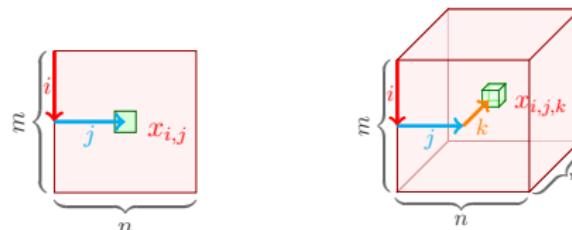
<https://xinychen.github.io/papers/thesis.pdf>

About me:

- Homepage: <https://xinychen.github.io>
- GitHub: <https://github.com/xinychen>
- How to reach me: chenxy346@gmail.com

What Is Tensors?

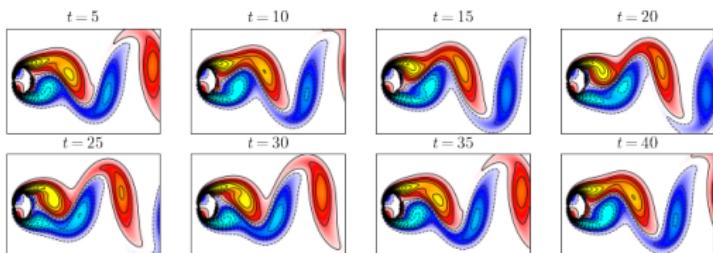
- What is tensor? $\mathbf{X} \in \mathbb{R}^{m \times n}$ vs. $\mathcal{X} \in \mathbb{R}^{m \times n \times t}$



- Tensors are everywhere!



Color image with
RGB channels



Dynamical system (fluid flow)

Nonstationary Temporal Matrix Factorization

Rewrite VAR in the form of matrix

Temporal operators

For any multivariate time series $\mathbf{X} \in \mathbb{R}^{R \times T}$ with $m, d \in \mathbb{N}^+$, if we define temporal operators as

$$\begin{aligned}\Psi_k &\triangleq \begin{bmatrix} \mathbf{0}_{(T-d-m) \times (d-k)} & -\mathbf{I}_{T-d-m} & \mathbf{0}_{(T-d-m) \times (k+m)} \end{bmatrix} \\ &+ \begin{bmatrix} \mathbf{0}_{(T-d-m) \times (d+m-k)} & \mathbf{I}_{T-d-m} & \mathbf{0}_{(T-d-m) \times k} \end{bmatrix} \\ &\in \mathbb{R}^{(T-d-m) \times T}, \quad k = 0, 1, \dots, d\end{aligned}$$

then

$$\begin{aligned}&\sum_{t=d+m+1}^T \|(\mathbf{x}_t - \mathbf{x}_{t-m}) - \sum_{k=1}^d \mathbf{A}_k (\mathbf{x}_{t-k} - \mathbf{x}_{t-m-k})\|_2^2 \\ &\equiv \|\mathbf{X} \Psi_0^\top - \sum_{k=1}^d \mathbf{A}_k \mathbf{X} \Psi_k^\top\|_F^2 \triangleq \|\mathbf{X} \Psi_0^\top - \mathbf{A} (\mathbf{I}_d \otimes \mathbf{X}) \Psi^\top\|_F^2\end{aligned}$$

where $\mathbf{A} \triangleq [\mathbf{A}_1 \quad \cdots \quad \mathbf{A}_d]$ and $\Psi \triangleq [\Psi_1 \quad \cdots \quad \Psi_d]$.

Nonstationary Temporal Matrix Factorization

Rewrite NoTMF:

$$\begin{aligned} \min_{\mathbf{W}, \mathbf{X}, \mathbf{A}} \quad & \frac{1}{2} \|\mathcal{P}_\Omega(\mathbf{Y} - \mathbf{W}^\top \mathbf{X})\|_F^2 + \frac{\rho}{2} (\|\mathbf{W}\|_F^2 + \|\mathbf{X}\|_F^2) \\ & + \frac{\gamma}{2} \|\mathbf{X} \Psi_0^\top - \mathbf{A} (\mathbf{I}_d \otimes \mathbf{X}) \Psi^\top\|_F^2 \end{aligned}$$

Alternating minimization method:

- w.r.t. \mathbf{W} :

$$\frac{\partial f}{\partial \mathbf{W}} = -\mathbf{X} \mathcal{P}_\Omega^\top (\mathbf{Y} - \mathbf{W}^\top \mathbf{X}) + \rho \mathbf{W} = \mathbf{0} \quad (\text{Least squares})$$

- w.r.t. \mathbf{X} :

$$\frac{\partial f}{\partial \mathbf{X}} = -\mathbf{W} \mathcal{P}_\Omega (\mathbf{Y} - \mathbf{W}^\top \mathbf{X}) + \rho \mathbf{X} + \gamma \sum_{k=0}^d \mathbf{A}_k^\top \left(\sum_{h=0}^d \mathbf{A}_h \mathbf{X} \Psi_h^\top \right) \Psi_k = \mathbf{0}$$

This generalized Sylvester equation can be solved by **conjugate gradient**.

- w.r.t. \mathbf{A} :

$$\mathbf{A} = \mathbf{X} \Psi_0^\top [(\mathbf{I}_d \otimes \mathbf{X}) \Psi^\top]^\dagger \quad (\text{Least squares})$$

Low-Rank Autoregressive Tensor Completion

- Augmented Lagrangian function:

$$\mathcal{L}(\mathbf{X}, \mathbf{Z}, \mathbf{W}) = \|\mathbf{X}\|_{r,*} + \frac{\gamma}{2} \|\mathbf{Z}\|_{\mathbf{A}, \mathcal{H}} + \frac{\lambda}{2} \|\mathbf{X} - \mathcal{Q}(\mathbf{Z})\|_F^2 + \langle \mathbf{W}, \mathbf{X} - \mathcal{Q}(\mathbf{Z}) \rangle + \pi(\mathbf{Z})$$

- The ADMM⁷ scheme:

$$\begin{cases} \mathbf{X} := \arg \min_{\mathbf{X}} \mathcal{L}(\mathbf{X}, \mathbf{Z}, \mathbf{W}) & \text{(Truncated nuclear norm minimization)} \\ \mathbf{Z} := \arg \min_{\mathbf{Z}} \mathcal{L}(\mathbf{X}, \mathbf{Z}, \mathbf{W}) & \text{(Generalized Sylvester equation)} \\ \mathbf{W} := \mathbf{W} + \lambda(\mathbf{X} - \mathcal{Q}(\mathbf{Z})) & \text{(Standard update)} \end{cases}$$

- ✓ Solution to \mathbf{X} : singular value thresholding
- ✓ Solution to \mathbf{Z} : conjugate gradient

⁷Alternating Direction Method of Multipliers.

Convolution & FFT

Flipping Operation in LCR

Results on speed fields

Training, Validation & Testing

Tuning Hyperparameters