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

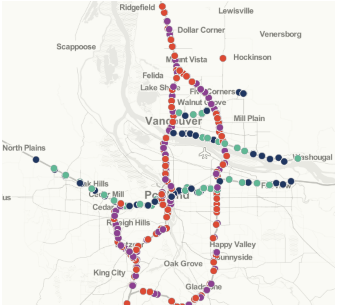
Outline

1. **Background**
2. **Literature Review**
3. **Nonstationary Temporal Matrix Factorization**
4. **Low-Rank Autoregressive Tensor Completion**
5. **Laplacian Convolutional Representation**
6. **Hankel Tensor Factorization**
7. **Experiments**
8. **Conclusion**

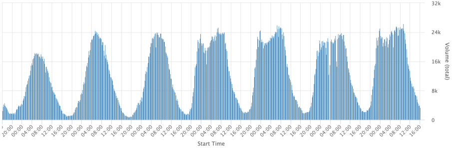
Multivariate Traffic Time Series

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

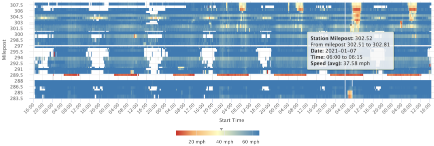
- **Example**: Portland highway traffic data[1].



Highway network & sensor locations



Traffic volume



Traffic speed

- $\boldsymbol{X} \in \mathbb{R}^{N \times T}$ 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.

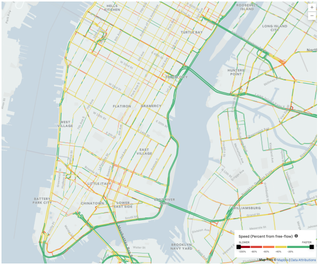
- Incompleteness & sparsity
- High-dimensionality
- Multidimensionality
- Noises & outliers
- Nonstationarity
- ......

In addition, spatiotemporal correlations are also very important.

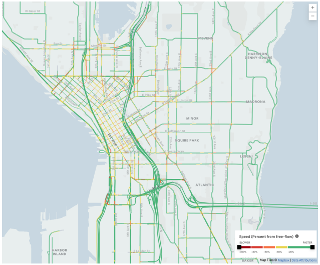
Multiple Data Behaviors

Sparsity & high-dimensionality

- Uber (hourly) movement speed data



NYC movement



Seattle movement

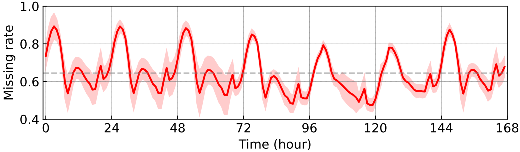
- {(road segment, time slot (hour)), average speed}
- Computing hourly speed: Road segments have 5+ unique trips.

Issue: Insufficient sampling of ridesharing vehicles on the road network!

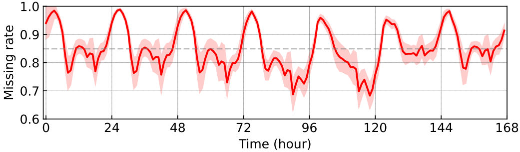
Multiple Data Behaviors

Sparsity & high-dimensionality

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

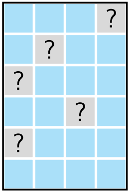


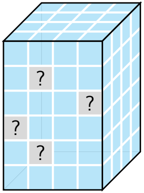
- **Seattle** movement speed data (2019)
  - **63,490** road segments & 8,760 time steps (hours)
  - Overall missing rate: **84.95%**



Problem Formulation

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



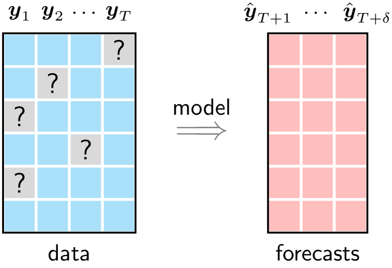


[Q]

- How to reconstruct missing values from observed data?
  - Matrix completion: From $\mathcal{P}_{\Omega}(\boldsymbol{Y})$ (observed) to $\mathcal{P}_{\Omega}^{\perp}(\boldsymbol{Y})$ (unobserved)
  - Tensor completion: From $\mathcal{P}_{\Omega}(\boldsymbol{\mathcal{Y}})$ (observed) to $\mathcal{P}_{\Omega}^{\perp}(\boldsymbol{\mathcal{Y}})$ (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 $\boldsymbol{Y}\in\mathbb{R}^{N\times T}$ consisting of time series $\boldsymbol{y}_1,\ldots,\boldsymbol{y}_T\in\mathbb{R}^N$, forecast data points $\boldsymbol{y}_{T+\delta},\delta\in\mathbb{N}^+$.

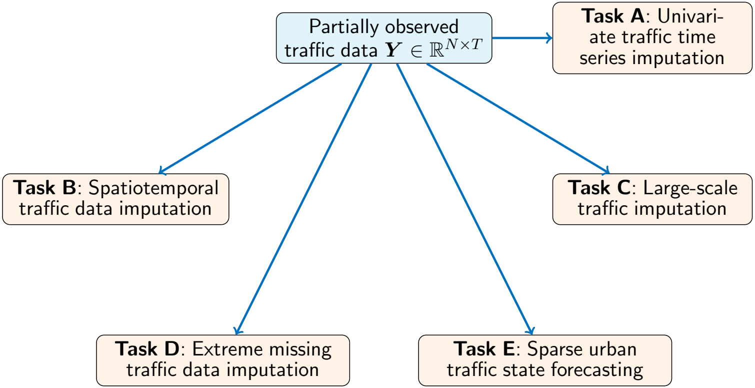


[Q]

- 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 imputation and forecasting**.



Imputation & Forecasting

Traffic data imputation

- Time series autoregression (Schafer'97, Chen & Shao'00)
- Principal component analysis (Qu et al.'09, Li et al.'13)
- Matrix factorization (Asif et al.'13, Asif et al.'16)
- Tensor factorization (Tan et al.'13, Chen et al.'19)
- Low-rank tensor completion (Ran et al.'16, Chen et al.'20)
- Temporal matrix/tensor factorization (Chen & Sun'22)

Time series forecasting on sparse data

- Autoregression predictor (Anava et al.'15)
- Prediction on the imputed data (e.g., Che et al.'18)
- Dynamic tensor completion (Tan et al.'16)
- Temporal matrix factorization (Yu et al.'16, Chen & Sun'22)
- Online matrix factorization (Gultekin & Paisley'18)

Tensor Factorization

- Revisit tensor factorization

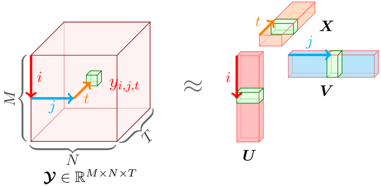
1927	1960s	1970	2009	2011
Higher-Order SVD	Tucker Decomposition	CP Decomposition	Tensor Decompositions and Applications	Tensor-Train Decomposition
Frank Lauren Hitchcock	Ledyard R. Tucker	J. Douglas Carroll Jih-Jie Chang Richard A. Harshman	Tamara G. Kolda	Ivan Oseledets

Frank Lauren Hitchcock

Tamara G. Kolda

Ivan Oseledets

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



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

Matrix/Tensor Completion

Matrix nuclear norm ($\|\boldsymbol{X}\|_*$) minimization
Candès & Recht'09

Singular value thresholding
Cai et al.'10

Truncated nuclear norm ($\|\boldsymbol{X}\|_{r,*}, r \in \mathbb{N}^+$) minimization
Zhang et al.'12
Hu et al.'12

Tensor nuclear norm ($\|\boldsymbol{\mathcal{X}}\|_*$) minimization
Liu et al.'13

Generalized nonconvex nonsmooth low-rank minimization
Lu et al.'14

Tensor nuclear norm minimization with linear transform
Lu et al.'19

Low-rank Hankel matrix/tensor ($\mathcal{H}_\tau(\cdot)$) completion
Yokota et al.'18
Sedighin et al.'20
Cai et al.'21
Yamamoto et al.'22

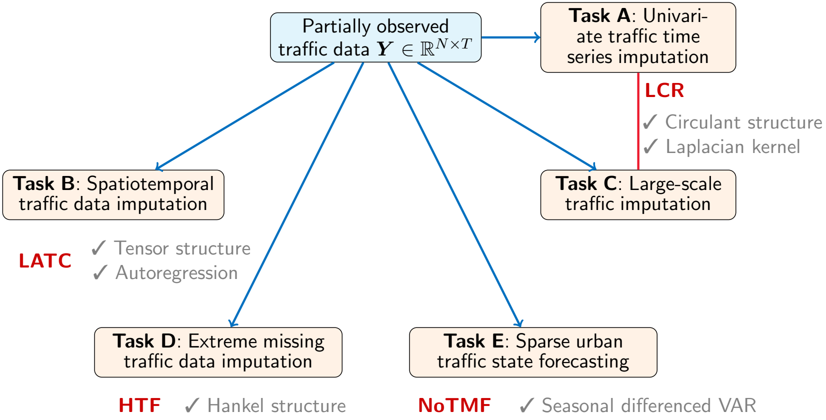
Circulant/Convolution nuclear norm ($\|\mathcal{C}(\boldsymbol{X})\|_*$ or $\|\mathcal{C}_{\tilde{\tau}}(\boldsymbol{X})\|_*$) minimization
Liu'22
Liu & Zhang'23

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

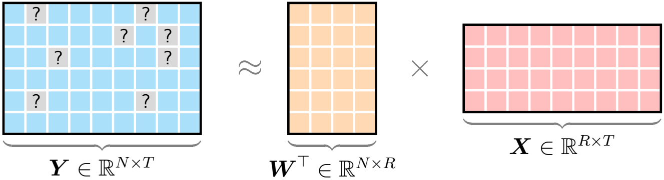
Overview

We are working on **spatiotemporal traffic data imputation and forecasting**.



Matrix Factorization

A simple approach to reconstruct missing values.



MF (Koren et al.'09)

Estimating low-dimensional $\boldsymbol{W}, \boldsymbol{X}$:

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

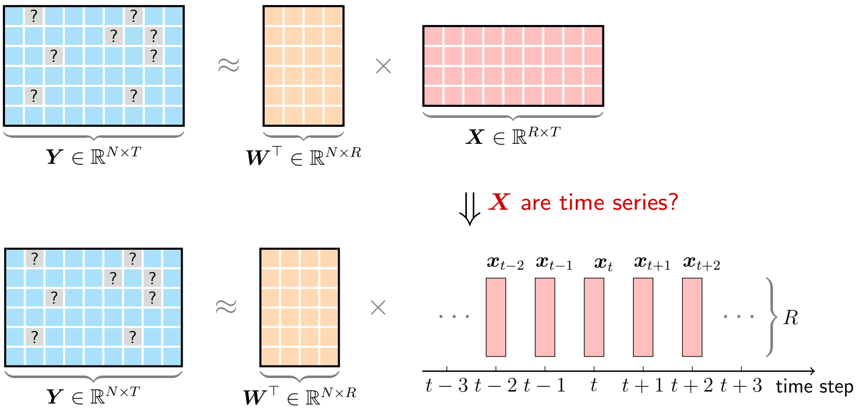
on data $\boldsymbol{Y}$ w/ observed index set Ω.

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

How to build temporal correlations on MF?

Temporal Matrix Factorization

Vector autoregression (VAR) on the temporal factor matrix.



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

Temporal Matrix Factorization

MF (Koren et al.'09)

Estimating low-dimensional $\boldsymbol{W},\boldsymbol{X}$:

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

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

$+$

dth-order VAR

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

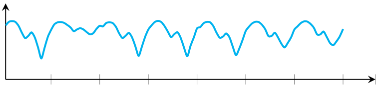
w/ coefficients $\{\boldsymbol{A}_k\}$.

$\Downarrow$ Yu et al.'16, Chen & Sun'21

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

Nonstationary Temporal Matrix Factorization

Traffic data are nonstationary due to daily patterns of traffic flow.



- Season-m differencing ($m \in \mathbb{N}^+$, e.g., daily/weekly):

$$\boldsymbol{x}_t \approx \sum_{k=1}^{d} \boldsymbol{A}_k \boldsymbol{x}_{t-k} \quad \Rightarrow \quad \boldsymbol{x}_t - \boldsymbol{x}_{t-m} \approx \sum_{k=1}^{d} \boldsymbol{A}_k (\boldsymbol{x}_{t-k} - \boldsymbol{x}_{t-m-k})$$

- **(Ours)** Optimization problem:

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

Nonstationary Temporal Matrix Factorization

Rewrite NoTMF

- Optimization problem:

$$\min_{\boldsymbol{W},\boldsymbol{X},\boldsymbol{A}} \ \frac{1}{2}\underbrace{\|\mathcal{P}_{\Omega}(\boldsymbol{Y}-\boldsymbol{W}^{\top}\boldsymbol{X})\|_F^2}_{\text{MF on data } \boldsymbol{Y}}+\frac{\rho}{2}\underbrace{(\|\boldsymbol{W}\|_F^2+\|\boldsymbol{X}\|_F^2)}_{\text{Regularization}}$$
$$+\frac{\gamma}{2}\underbrace{\|\boldsymbol{X}\boldsymbol{\Psi}_0^{\top}-\boldsymbol{A}(\boldsymbol{I}_d\otimes\boldsymbol{X})\boldsymbol{\Psi}^{\top}\|_F^2}_{\text{Temporal modeling on } \boldsymbol{X}}$$

where $\boldsymbol{\Psi}_0\in\mathbb{R}^{(T-d-m)\times T}$ and $\boldsymbol{\Psi}\in\mathbb{R}^{(T-d-m)\times(dT)}$ are temporal operators.

- Alternating minimization (let f be the obj.):

$$\begin{cases} \text{Spatial factors} & \boldsymbol{W}:=\{\boldsymbol{W}\mid\frac{\partial f}{\partial \boldsymbol{W}}=\boldsymbol{0}\} & \text{(least squares)} \\ \text{Temporal factors} & \boldsymbol{X}:=\{\boldsymbol{X}\mid\frac{\partial f}{\partial \boldsymbol{X}}=\boldsymbol{0}\} & \text{(conjugate gradient)} \\ \text{VAR coefficients} & \boldsymbol{A}:=\{\boldsymbol{A}\mid\frac{\partial f}{\partial \boldsymbol{A}}=\boldsymbol{0}\} & \text{(least squares)} \end{cases}$$

Nonstationary Temporal Matrix Factorization

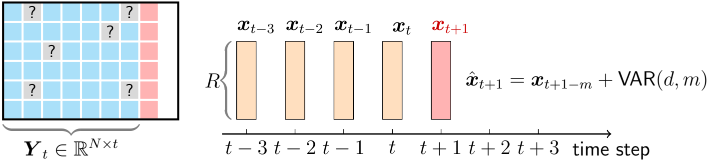
NoTMF forecasting on streaming data?

Implementation

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

✓ Sparse input $\boldsymbol{Y}_t$

✓ Forecast in latent spaces



Nonstationary Temporal Matrix Factorization

NoTMF forecasting on streaming data?

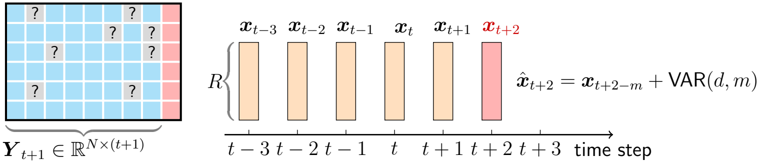
- Online forecasting (Gultekin & Paisley'18):
  - Fix the spatial factor matrix $\boldsymbol{W}$
  - Use input data $\boldsymbol{Y}_{t+1}$ to update the temporal factor matrix $\boldsymbol{X}$ and the coefficient matrix $\boldsymbol{A}$

Implementation

- Estimate $\boldsymbol{X}, \boldsymbol{A}$
- Forecast $\hat{\boldsymbol{x}}_{t+2}$ with VAR
- Return $\hat{\boldsymbol{y}}_{t+2} = \boldsymbol{W}^\top \hat{\boldsymbol{x}}_{t+2}$

✓ Sparse input $\boldsymbol{Y}_{t+1}$

✓ Forecast in latent spaces



Matrix/Tensor Completion

Cornerstone: Nuclear norm minimization in matrix/tensor completion

LRMC (Candès & Recht'09)

Estimating the matrix $\boldsymbol{X}$:

$$\min_{\boldsymbol{X}} \ \|\boldsymbol{X}\|_* \quad \text{s.t. } \mathcal{P}_\Omega(\boldsymbol{X}) = \mathcal{P}_\Omega(\boldsymbol{Y})$$

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

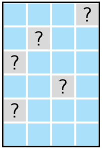
vs.

LRTC (Liu et al.'13)

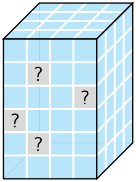
Estimating the tensor $\boldsymbol{\mathcal{X}}$:

$$\min_{\boldsymbol{\mathcal{X}}} \ \|\boldsymbol{\mathcal{X}}\|_* \quad \text{s.t. } \mathcal{P}_\Omega(\boldsymbol{\mathcal{X}}) = \mathcal{P}_\Omega(\boldsymbol{\mathcal{Y}})$$

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



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

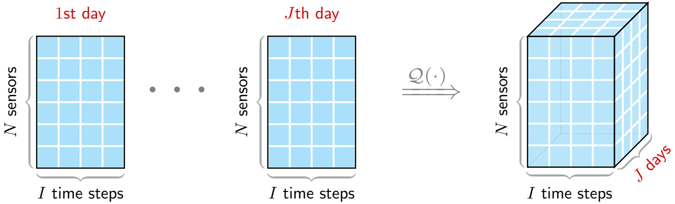


$\mathcal{P}_\Omega(\boldsymbol{\mathcal{Y}}) \in \mathbb{R}^{N\times I\times J}$

- **Limitation**: Only cover global consistency

Low-Rank Autoregressive Tensor Completion

- Introduce traffic tensors with day dimension[2] (Tan et al.'13, Chen et al.'19, ...)



[2] There are $T = IJ$ time steps in total.

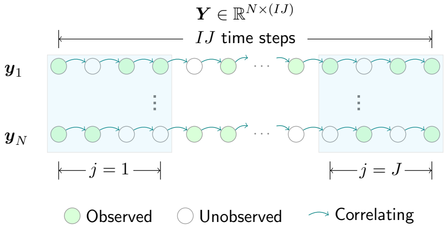
Low-Rank Autoregressive Tensor Completion

- Build temporal correlations with autoregression

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

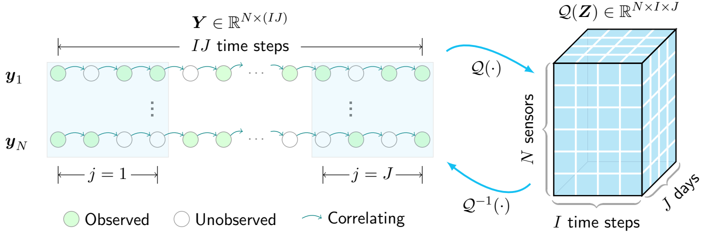
$$\boxed{\|\boldsymbol{Y}\|_{\boldsymbol{A},\mathcal{H}} \triangleq \sum_{n,t}\Big(y_{n,t} - \sum_{k} a_{n,k} y_{n,t-h_k}\Big)^2}$$

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



Low-Rank Autoregressive Tensor Completion

Local consistency w/ autoregression **Global consistency** w/ tensor structure



LATC

Optimization problem:

$$\min_{\boldsymbol{Z},\boldsymbol{A}} \; \|\mathcal{Q}(\boldsymbol{Z})\|_{r,*} + \frac{\gamma}{2}\|\boldsymbol{Z}\|_{\boldsymbol{A},\mathcal{H}}$$
$$\text{s.t. } \mathcal{P}_\Omega(\boldsymbol{Z}) = \mathcal{P}_\Omega(\boldsymbol{Y})$$

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

$\Rightarrow$

Two subproblems

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

Low-Rank Autoregressive Tensor Completion

$\boldsymbol{Z}$-subproblem:

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

- Augmented Lagrangian function:[3]

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

Implementation

Repeat
- Compute $\boldsymbol{Z}$
- Compute $\boldsymbol{A}$

$\Longrightarrow$

Implementation

Repeat
- Repeat
  # Alternating Direction Method of Multipliers (ADMM)
  - Compute $\boldsymbol{\mathcal{X}}$
  - Compute $\boldsymbol{Z}$
  - Compute $\boldsymbol{\mathcal{W}}$
- Compute $\boldsymbol{A}$

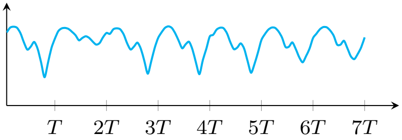
[3] $\boldsymbol{\mathcal{W}} \in \mathbb{R}^{N\times I\times J}$ (Lagrange multiplier); $\langle\cdot,\cdot\rangle$ (inner product). The indicator function:

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

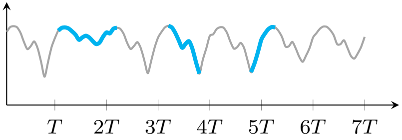
Laplacian Convolutional Representation

Motivation: Time series imputation

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



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

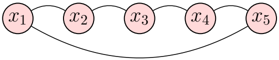


- **[Question]** How to characterize both global and local trends in sparse time series data?

Laplacian Convolutional Representation

Local trend modeling

- Intuition of (circulant) Laplacian matrix



Undirected and circulant graph

$$\xrightarrow{\text{Modeling}} \quad \boldsymbol{L}=\begin{bmatrix} 2 & -1 & 0 & 0 & -1 \\ -1 & 2 & -1 & 0 & 0 \\ 0 & -1 & 2 & -1 & 0 \\ 0 & 0 & -1 & 2 & -1 \\ -1 & 0 & 0 & -1 & 2 \end{bmatrix}$$

(Circulant) Laplacian matrix

- Define Laplacian kernel:

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

$$\Downarrow$$

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

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

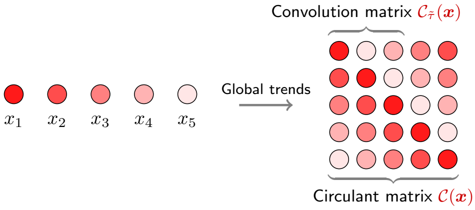
- (Laplacian) Temporal regularization:

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

Reformulate temporal regularization with circular convolution

Laplacian Convolutional Representation

Global trend modeling: Circulant matrix $\mathcal{C}(\boldsymbol{x})$ vs. convolution matrix $\mathcal{C}_{\tilde{\tau}}(\boldsymbol{x})$



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

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

Estimating $\boldsymbol{x}$:

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

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

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

Estimating $\boldsymbol{x}$:

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

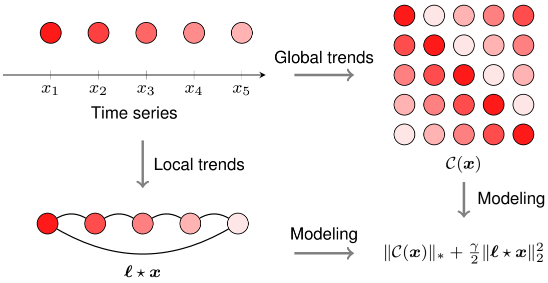
on data $\boldsymbol{y}$ w/ observed index set Ω.

Laplacian Convolutional Representation

Laplacian Convolutional Representation (LCR)

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

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



Laplacian Convolutional Representation

- Augmented Lagrangian function:[4]

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

- The ADMM scheme:

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

- Optimize $\boldsymbol{x}$?

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

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

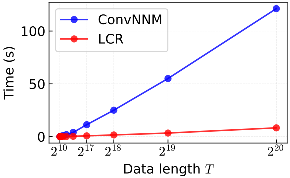
[4] $\boldsymbol{w}\in\mathbb{R}^T$ (Lagrange multiplier); $\langle\cdot,\cdot\rangle$ (inner product).

Laplacian Convolutional Representation

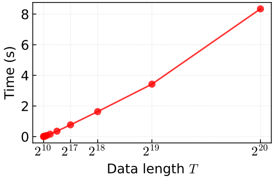
Empirical time complexity

On the synthetic data $\boldsymbol{y}\in\mathbb{R}^{T}$ 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**[5] (Liu'22, Liu & Zhang'23)
  - Convolution matrix $\mathcal{C}_{\tilde{\tau}}(\boldsymbol{y})\in\mathbb{R}^{T\times\tilde{\tau}}$ with kernel size $\tilde{\tau}=2^4$
  - Singular value thresholding in $\mathcal{O}(\tilde{\tau}^2T)$



ConvNNM vs. LCR



LCR

[5] Convolution nuclear norm minimization.

Laplacian Convolutional Representation

LCR

On time series $\boldsymbol{y} \in \mathbb{R}^{T}$,

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

w/ observed index set Ω.

$\Longrightarrow$

LCR-2D

On time series $\boldsymbol{Y} \in \mathbb{R}^{N \times T}$,

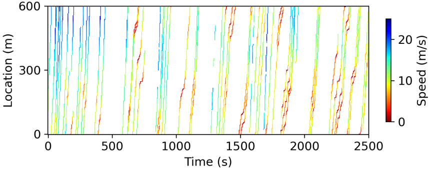
$$\begin{aligned}\min_{\boldsymbol{X}}\ & \|\mathcal{C}(\boldsymbol{X})\|_* + \frac{\gamma}{2}\|(\boldsymbol{\ell}_s\boldsymbol{\ell}^\top) \star \boldsymbol{X}\|_F^2 \\ \text{s.t. } & \|\mathcal{P}_{\Omega}(\boldsymbol{X}-\boldsymbol{Y})\|_F \leq \epsilon\end{aligned}$$

w/ observed index set Ω.

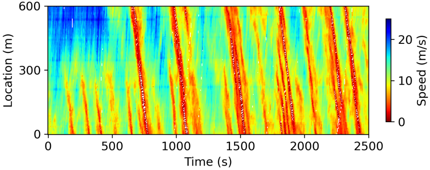
Hankel Tensor Factorization

Motivation: Spatiotemporal data reconstruction

- Speed field reconstruction problem in vehicular traffic flow.



200-by-500 matrix (NGSIM) $\Downarrow$ Reconstruct speed field from 5% sparse trajectories?



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

Hankel Tensor Factorization

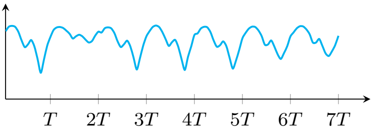
- Hankel matrix
  - Given $\boldsymbol{x} = (1, 2, 3, 4, 5)^\top$ and window length $\tau = 2$, we have

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

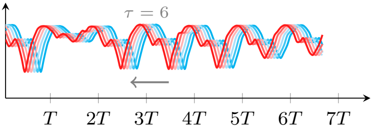
Hankel matrix (Source: Twitter)

Hankel Tensor Factorization

- Hankel matrix
  - Automatic temporal modeling



Traffic time series



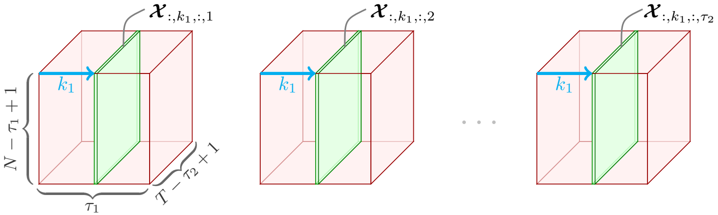
Hankel matrix

Hankel Tensor Factorization

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

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

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

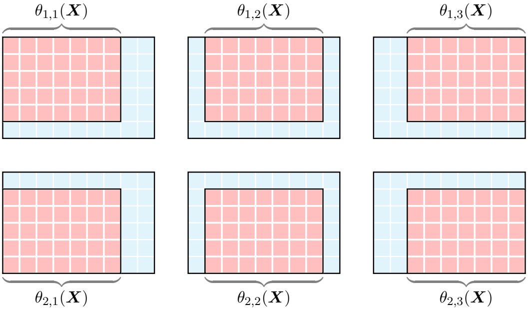
Hankel Tensor Factorization

Hankel indexing:

- Sampling function for the Hankelization:

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

  referring to the tensor slice with $k_1 \in \{1,\ldots,\tau_1\}$, $k_2 \in \{1,\ldots,\tau_2\}$.
- **[Importance]** Developing memory-efficient algorithms.



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

Hankel Tensor Factorization

Ours:

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

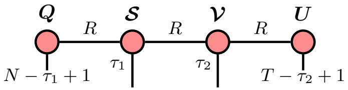
$$\theta_{k_1,k_2}(\boldsymbol{Y}) \approx (\boldsymbol{Q} \star_{\text{row}} \boldsymbol{s}_{k_1}^\top)(\boldsymbol{U} \star_{\text{row}} \boldsymbol{v}_{k_2}^\top)^\top$$

Baselines:

- Tensor-train decomposition:

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

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



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

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

Hankel Tensor Factorization

HTF (convolutional decomposition)

- Optimization problem:

$$\min_{\boldsymbol{Q},\boldsymbol{S},\boldsymbol{U},\boldsymbol{V}}\ \frac{1}{2}\underbrace{\sum_{k_1,k_2}\left\|\mathcal{P}_{\Omega_{k_1,k_2}}(\theta_{k_1,k_2}(\boldsymbol{Y})-(\boldsymbol{Q}\star_{\text{row}}\boldsymbol{s}_{k_1})(\boldsymbol{U}\star_{\text{row}}\boldsymbol{v}_{k_2})^\top)\right\|_F^2}_{\text{Tensor decomposition on Hankel tensor slices}}$$

$$+\frac{\rho}{2}(\|\boldsymbol{Q}\|_F^2+\|\boldsymbol{S}\|_F^2+\|\boldsymbol{U}\|_F^2+\|\boldsymbol{V}\|_F^2)$$

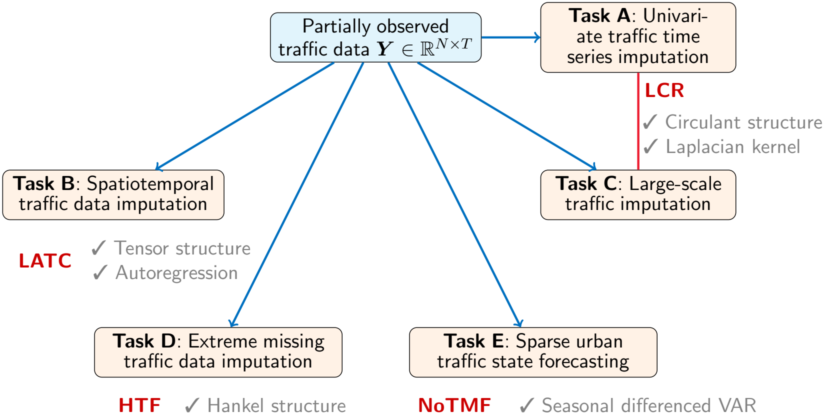
- Alternating minimization (let f be the obj.):

$$\begin{cases}\boldsymbol{Q}:=\{\boldsymbol{Q}\mid\frac{\partial f}{\partial\boldsymbol{Q}}=\boldsymbol{0}\} & \text{(conjugate gradient)}\\ \boldsymbol{s}_{k_1}:=\{\boldsymbol{s}_{k_1}\mid\frac{\partial f}{\partial\boldsymbol{s}_{k_1}}=\boldsymbol{0}\},\ \forall k_1 & \text{(conjugate gradient)}\\ \boldsymbol{U}:=\{\boldsymbol{U}\mid\frac{\partial f}{\partial\boldsymbol{U}}=\boldsymbol{0}\} & \text{(conjugate gradient)}\\ \boldsymbol{v}_{k_2}:=\{\boldsymbol{v}_{k_2}\mid\frac{\partial f}{\partial\boldsymbol{v}_{k_2}}=\boldsymbol{0}\},\ \forall k_2 & \text{(conjugate gradient)}\end{cases}$$

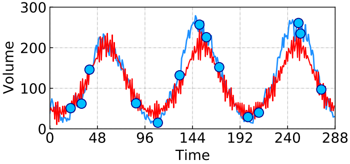
- Memory-efficient but still computationally costly!

Overview

We are working on **spatiotemporal traffic data imputation and forecasting**.



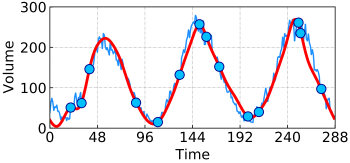
Univariate Traffic Time Series Imputation



CircNNM:

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

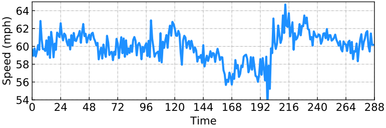
$\Downarrow$ Plus temporal regularization



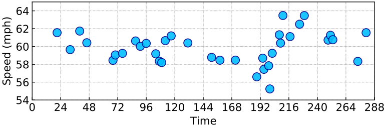
LCR:

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

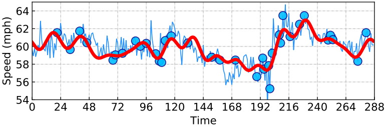
Univariate Traffic Time Series Imputation



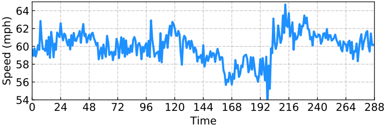
Mask **90%** observations



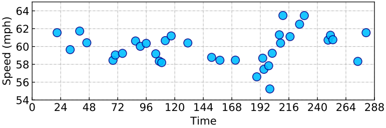
Reconstruct time series



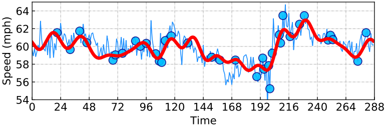
Univariate Traffic Time Series Imputation

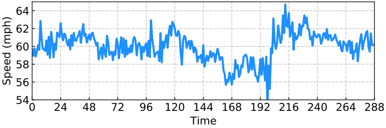


Mask **90%** observations

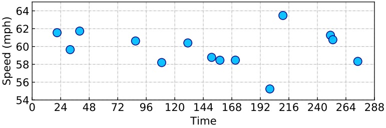


Reconstruct time series

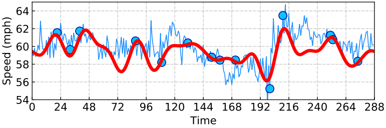




Mask **95%** observations



Reconstruct time series



Spatiotemporal Traffic Data Imputation

LATC vs. baseline models (in MAPE/RMSE)

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

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 ($\boldsymbol{Y} \in \mathbb{R}^{1156\times 2976}$)

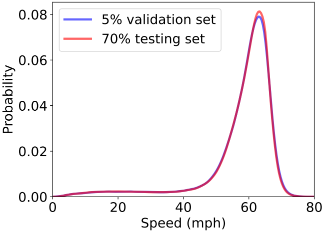
Missing rate	LATC	LAMC	LRTC-TNN	BTMF	SPC
30%, Random Missing	**16.95**/**15.99**	17.93/16.03	17.27/16.08	18.22/19.14	21.29/56.73
70%, Random Missing	**19.59**/**18.70**	21.26/19.37	19.99/18.73	19.96/22.21	24.35/43.32
90%, Random Missing	23.15/22.83	25.64/23.75	**22.90**/**22.68**	23.90/25.71	28.45/39.65
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

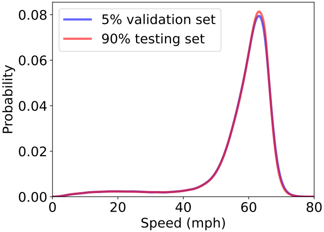
- LATC vs. LAMC: The significance of tensor representation
- LATC vs. LRTC-TNN: The significance of temporal autoregression

Spatiotemporal Traffic Data Imputation

Parameter tuning process: Training set, validation set, and testing set?

- Random missing on the Seattle freeway traffic speed dataset





- Imputation performance (e.g., 70% missing rate)

On the validation set (5% data)

γ/λ	Truncation parameter				
	$r=5$	$r=10$	$r=15$	$r=20$	$r=25$
1/10	7.84/4.52	7.20/4.25	6.82/4.08	6.60/3.98	6.41/3.92
1/5	7.84/4.52	7.20/4.25	6.82/4.08	6.59/3.97	6.41/3.92
1	7.81/4.51	7.18/4.24	6.80/4.07	6.58/3.97	6.39/3.91
5	7.70/4.45	7.09/4.20	6.72/4.04	6.49/3.93	6.29/3.87
10	7.59/4.39	7.00/4.16	6.64/4.00	6.41/3.89	**6.22/3.83**

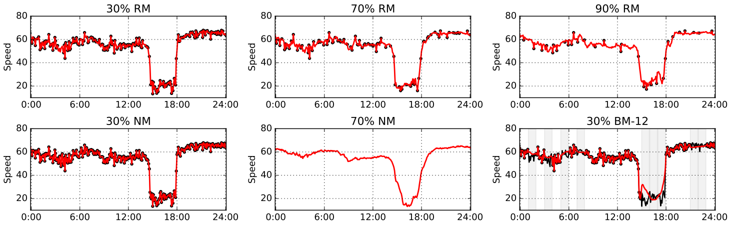
On the testing set (70% data)

γ/λ	Truncation parameter				
	$r=5$	$r=10$	$r=15$	$r=20$	$r=25$
1/10	7.83/4.53	7.18/4.27	6.80/4.09	6.58/3.99	6.41/3.92
1/5	7.83/4.53	7.18/4.26	6.80/4.09	6.57/3.98	6.40/3.92
1	7.80/4.52	7.16/4.25	6.78/4.08	6.55/3.98	6.40/3.92
5	7.70/4.47	7.08/4.21	6.70/4.04	6.46/3.94	6.29/3.87
10	7.58/4.41	6.99/4.17	6.62/4.01	6.39/3.90	**6.21/3.84**

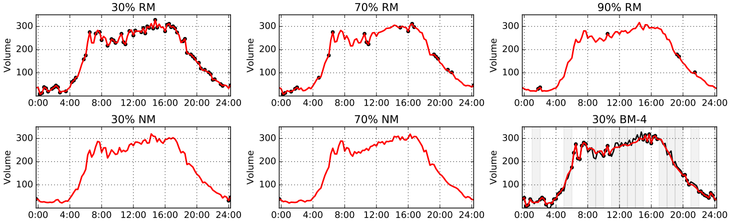
Spatiotemporal Traffic Data Imputation

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 ($\boldsymbol{Y} \in \mathbb{R}^{11160 \times 8064}$)

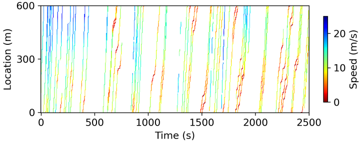
Model	Missing rate			
	30%	50%	70%	90%
LCR-2D	1.50/**1.49**	1.76/**1.69**	**2.07**/**2.06**	**3.19**/**3.05**
LCR_N	**1.48**/1.50	**1.73**/1.73	**2.07**/2.12	3.24/3.22
LCR	1.50/**1.49**	1.76/**1.69**	2.08/2.07	3.21/3.06
CTNNM	2.26/1.84	2.67/2.14	3.40/2.66	5.22/3.90
CircNNM	2.26/1.84	2.69/2.15	3.43/2.67	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

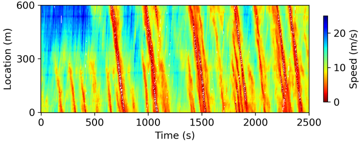
- LCR-2D $>$ CTNNM: The importance of temporal regularization.
- CTNNM $\geq$ CircNNM: Ciculant tensor is superior to circulant matrix.
- LCR $>$ LRMC/LRTC: The importance of global/local modeling.

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

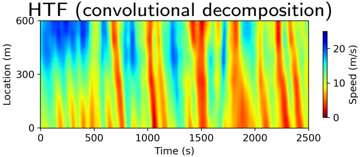
Extreme Missing Traffic Data Imputation



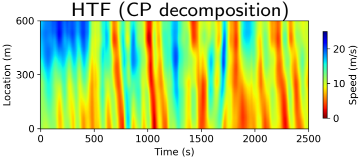
Sparse speed field



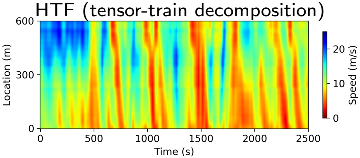
Ground truth speed field



MAPE = **51.92%**



MAPE = 53.93%



MAPE = 56.48%

Extreme Missing Traffic Data Imputation

HTF vs. baseline models (in MAPE/RMSE)

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

Model	Random missing rate			
	80%	85%	90%	95%
HTF (`convolution`)	**6.21/3.88**	**6.51/4.06**	**6.98/4.30**	**8.02/4.84**
HTF (`tensor-train`)	8.75/5.16	9.86/5.76	9.24/5.36	9.89/5.70
HTF (`CP`)	7.25/4.33	7.93/4.66	8.61/4.96	9.25/5.20
LATC	6.50/4.00	6.90/4.21	7.47/4.51	8.75/5.05
LRTC-TNN	6.97/4.24	7.43/4.43	8.19/4.81	9.60/5.55
LCR	6.75/4.15	7.31/4.38	7.96/4.71	9.78/5.39
BTMF	6.85/4.17	7.36/4.42	8.13/4.79	9.63/5.48

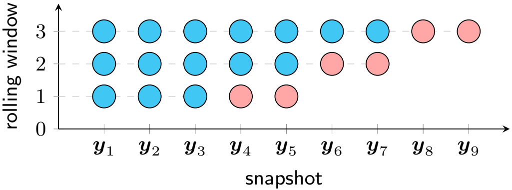
Results

- Convolutional tensor decomposition outperforms both tensor-train and CP tensor decomposition.
- Our HTF model performs better than state-of-the-art baseline models.

Sparse Urban Traffic State Forecasting

NoTMF forecasting

- NYC movement speed dataset:
  - Ten-week data of size 98210×1680
  - Contain 66.56% missing values
- Rolling forecasting setup:
  - Training set: 8-week data
  - Validation set: 1-week data
  - Testing set: 1-week data
  - Time horizon: $\delta = 1, 2, 3, 6$
- Rolling forecasting illustration ($\delta = 2$):



Sparse Urban Traffic State Forecasting

Sparse Urban Traffic State Forecasting

Conclusion

- Matrix nuclear norm ($\|\boldsymbol{X}\|_*$) minimization — Candès & Recht'09
- Singular value thresholding — Cai et al.'10
- Truncated nuclear norm ($\|\boldsymbol{X}\|_{r,*}, r \in \mathbb{N}^+$) minimization — Zhang et al.'12, Hu et al.'12
- Tensor nuclear norm ($\|\boldsymbol{\mathcal{X}}\|_*$) minimization — Liu et al.'13
- Generalized nonconvex nonsmooth low-rank minimization — Lu et al.'14
- Tensor nuclear norm minimization with linear transform — Lu et al.'19
- Low-rank Hankel matrix/tensor ($\mathcal{H}_\tau(\cdot)$) completion — Yokota et al.'18, Sedighin et al.'20, Cai et al.'21, Yamamoto et al.'22
- Circulant/Convolution nuclear norm ($\|\mathcal{C}(\boldsymbol{X})\|_*$ or $\|\mathcal{C}_{\tilde{\tau}}(\boldsymbol{X})\|_*$) minimization — Liu'22, Liu & Zhang'23

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 $\|\mathcal{Q}(\boldsymbol{\mathcal{Z}})\|_{r,*}$ ✓ Autoregression $\|\boldsymbol{Z}\|_{\boldsymbol{A},\mathcal{H}}$

Laplacian Convolutional Representation (LCR):

- (Solution) Global & local time series trends
  - Global trend modeling: $\|\mathcal{C}(\boldsymbol{x})\|_*$
  - Local trend modeling: $\|\boldsymbol{\ell}\star\boldsymbol{x}\|_2^2$
- (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. 2009, 9(6): 717-772.
- (Cai et al.'10) "A singular value thresholding algorithm for matrix completion." SIAM Journal on optimization. 2010, 20(4): 1956-1982.
- (Zhang et al.'12) "Matrix completion by truncated nuclear norm regularization." IEEE Conference on computer vision and pattern recognition. 2012.
- (Hu et al.'12) "Fast and accurate matrix completion via truncated nuclear norm regularization." IEEE transactions on pattern analysis and machine intelligence. 2012, 35(9): 2117-2130.
- (Lu et al.'14) "Generalized nonconvex nonsmooth low-rank minimization." Proceedings of the IEEE conference on computer vision and pattern recognition. 2014.
- (Lu et al.'19) "Tensor robust principal component analysis with a new tensor nuclear norm." IEEE transactions on pattern analysis and machine intelligence. 2019, 42(4): 925-938.
- (Yokota et al.'18) "Missing slice recovery for tensors using a low-rank model in embedded space." Proceedings of the IEEE conference on computer vision and pattern recognition. 2018.
- (Cai et al.'21) "Accelerated structured alternating projections for robust spectrally sparse signal recovery." IEEE Transactions on Signal Processing. 2021, 69: 809-821.
- (Liu'22) "Time series forecasting via learning convolutionally low-rank models." IEEE Transactions on Information Theory. 2022, 68(5): 3362-3380.
- (Liu & Zhang'23) "Recovery of future data via convolution nuclear norm minimization." IEEE Transactions on Information Theory. 2023, 69(1): 650-665.

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

Nonstationary Temporal Matrix Factorization

Rewrite VAR in the form of matrix

Temporal operators

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

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

then

$$\sum_{t=d+m+1}^{T} \|(\boldsymbol{x}_t - \boldsymbol{x}_{t-m}) - \sum_{k=1}^{d} \boldsymbol{A}_k(\boldsymbol{x}_{t-k} - \boldsymbol{x}_{t-m-k})\|_2^2$$

$$\equiv \|\boldsymbol{X}\boldsymbol{\Psi}_0^\top - \sum_{k=1}^{d} \boldsymbol{A}_k \boldsymbol{X} \boldsymbol{\Psi}_k^\top\|_F^2 \triangleq \|\boldsymbol{X}\boldsymbol{\Psi}_0^\top - \boldsymbol{A}(\boldsymbol{I}_d \otimes \boldsymbol{X})\boldsymbol{\Psi}^\top\|_F^2$$

where $\boldsymbol{A} \triangleq \begin{bmatrix} \boldsymbol{A}_1 & \cdots & \boldsymbol{A}_d \end{bmatrix}$ and $\boldsymbol{\Psi} \triangleq \begin{bmatrix} \boldsymbol{\Psi}_1 & \cdots & \boldsymbol{\Psi}_d \end{bmatrix}$.

Nonstationary Temporal Matrix Factorization

Rewrite NoTMF:

$$\begin{aligned}\min_{\boldsymbol{W},\boldsymbol{X},\boldsymbol{A}}\ & \frac{1}{2}\|\mathcal{P}_{\Omega}(\boldsymbol{Y}-\boldsymbol{W}^\top\boldsymbol{X})\|_F^2+\frac{\rho}{2}(\|\boldsymbol{W}\|_F^2+\|\boldsymbol{X}\|_F^2)\\ & +\frac{\gamma}{2}\|\boldsymbol{X}\boldsymbol{\Psi}_0^\top-\boldsymbol{A}(\boldsymbol{I}_d\otimes\boldsymbol{X})\boldsymbol{\Psi}^\top\|_F^2\end{aligned}$$

Alternating minimization method:

- w.r.t. $\boldsymbol{W}$:

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

- w.r.t. $\boldsymbol{X}$:

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

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

- w.r.t. $\boldsymbol{A}$:

$$\boldsymbol{A}=\boldsymbol{X}\boldsymbol{\Psi}_0^\top[(\boldsymbol{I}_d\otimes\boldsymbol{X})\boldsymbol{\Psi}^\top]^\dagger\qquad\text{(Least squares)}$$

Low-Rank Autoregressive Tensor Completion

- Augmented Lagrangian function:

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

- The ADMM[6] scheme:

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

✓ Solution to $\boldsymbol{\mathcal{X}}$: singular value thresholding

✓ Solution to $\boldsymbol{Z}$: conjugate gradient

[6] Alternating Direction Method of Multipliers.

Laplacian Convolutional Representation

- Optimize $\boldsymbol{x}$ via FFT (in $\mathcal{O}(T\log T)$ time):

$$\boldsymbol{x} := \arg\min_{\boldsymbol{x}} \ \|\mathcal{C}(\boldsymbol{x})\|_{*} + \frac{\gamma}{2}\|\boldsymbol{\ell}\star\boldsymbol{x}\|_2^2 + \frac{\lambda}{2}\|\boldsymbol{x}-\boldsymbol{z}+\boldsymbol{w}/\lambda\|_2^2$$

$$\Longrightarrow \hat{\boldsymbol{x}} := \arg\min_{\hat{\boldsymbol{x}}} \ \|\hat{\boldsymbol{x}}\|_1 + \frac{\gamma}{2T}\|\hat{\boldsymbol{\ell}}\circ\hat{\boldsymbol{x}}\|_2^2 + \frac{\lambda}{2T}\|\hat{\boldsymbol{x}}-\hat{\boldsymbol{z}}+\hat{\boldsymbol{w}}/\lambda\|_2^2$$

where we introduce $\{\hat{\boldsymbol{\ell}},\hat{\boldsymbol{x}},\hat{\boldsymbol{z}},\hat{\boldsymbol{w}}\} \triangleq \mathcal{F}\{\boldsymbol{\ell},\boldsymbol{x},\boldsymbol{z},\boldsymbol{w}\}$ (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:

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

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

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

Flipping Operation in LCR

Results on speed fields

Tuning Hyperparameters