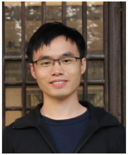
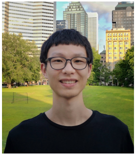
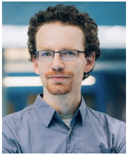
Laplacian Convolutional Representation for Traffic Time Series Imputation

Xinyu Chen

July 19, 2023







Ph.D. Candidate
Xinyu Chen
Polytechnique Montréal

Postdoc
Dr. Zhanhong Cheng
McGill University

Supervisor
Prof. Nicolas Saunier
Polytechnique Montréal

Co-supervisor
Prof. Lijun Sun
McGill University

Preprint:

- X. Chen, Z. Cheng, N. Saunier, L. Sun (2022). Laplacian convolutional representation for traffic time series imputation. arXiv preprint arXiv:2212.01529. https://xinychen.github.io/papers/Laplacian_convolution.pdf

GitHub repository:

- **transdim**: Machine learning for spatiotemporal traffic data imputation and forecasting. (1,000+ stars & 270+ forks on GitHub) https://github.com/xinychen/transdim

Slides:

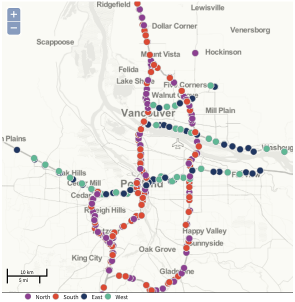
- https://xinychen.github.io/slides/LCR.pdf

Outline

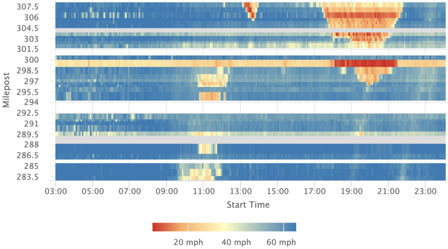
- **Motivation**
  - Data-Driven ITS
  - Time Series Imputation
  - Speed Field Reconstruction
- **Revisit Laplacian Matrix & Circular Convolution**
  - Laplacian Matrix
  - Laplacian Regularization
- **Circulant Matrix Nuclear Norm Minimization**
- **Laplacian Convolutional Representation**
  - Model Description
  - Solution Algorithm
- **Experiments**
  - Univariate Traffic Time Series Imputation
  - Speed Field Reconstruction
- **Conclusion**

Motivation

- Portland highway traffic flow data[1]



Highway network & sensor locations

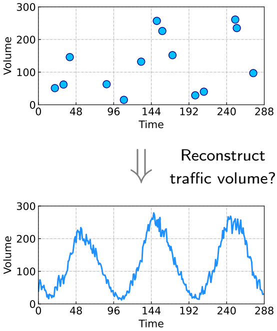


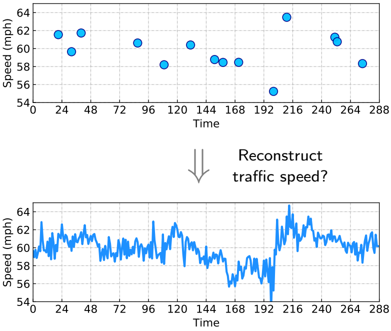
Traffic speed field

- Speed field $\boldsymbol{Y} \in \mathbb{R}^{N \times T}$ (N locations & T time steps)
- Speed field shows strong spatial/temporal dependencies

[1] https://portal.its.pdx.edu/home

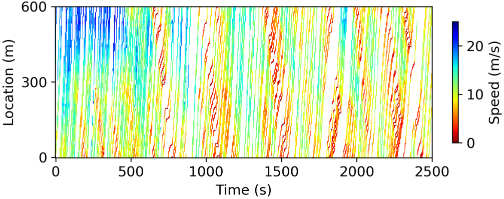
Motivation



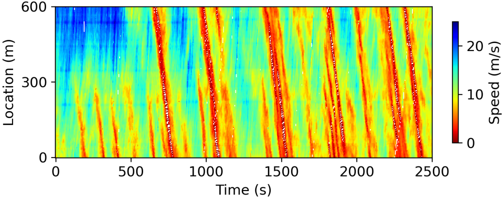


- How to utilize the global trends of traffic time series?
- How to produce local consistency of traffic data?

Motivation



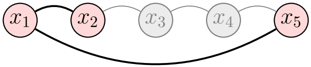
200-by-500 matrix (NGSIM) $\Downarrow$ Reconstruct speed field from 20% sparse trajectories?



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

Revisit Laplacian Matrix & Circular Convolution

- Intuition of (circulant) Laplacian matrix.



Undirected and circulant graph

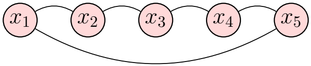
Modeling ⟶

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

(Circulant) Laplacian matrix

Revisit Laplacian Matrix & Circular Convolution

- Intuition of (circulant) Laplacian matrix.



Undirected and circulant graph

Modeling $\longrightarrow$

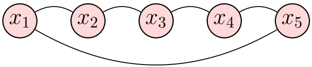
$$\boldsymbol{L} = \begin{bmatrix} \mathbf{2} & -1 & 0 & 0 & -1 \\ \mathbf{-1} & 2 & -1 & 0 & 0 \\ 0 & -1 & 2 & -1 & 0 \\ 0 & 0 & -1 & 2 & -1 \\ \mathbf{-1} & 0 & 0 & -1 & 2 \end{bmatrix}$$

(Circulant) Laplacian matrix

Revisit Laplacian Matrix & Circular Convolution

Reformulate Laplacian regularization with circular convolution.

- Intuition of (circulant) Laplacian matrix.



Undirected and circulant graph

Modeling $\longrightarrow$

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

(Circulant) Laplacian matrix

- Laplacian kernel: $\boldsymbol{\ell}=(2,-1,0,0,-1)^\top$.

$$\boldsymbol{L}\boldsymbol{x}=\begin{bmatrix}2 & -1 & 0 & 0 & -1\\ -1 & 2 & -1 & 0 & 0\\ 0 & -1 & 2 & -1 & 0\\ 0 & 0 & -1 & 2 & -1\\ -1 & 0 & 0 & -1 & 2\end{bmatrix}\begin{bmatrix}x_1\\ x_2\\ x_3\\ x_4\\ x_5\end{bmatrix}=\begin{bmatrix}2\\ -1\\ 0\\ 0\\ -1\end{bmatrix}\star\begin{bmatrix}x_1\\ x_2\\ x_3\\ x_4\\ x_5\end{bmatrix}=\boldsymbol{\ell}\star\boldsymbol{x}$$

where $\star$ denotes the ciruclar convolution.

- Local trend modeling via (Laplacian) temporal regularization:

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

Revisit Laplacian Matrix & Circular Convolution

Reformulate Laplacian regularization with circular convolution.

- Define Laplacian kernel:

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

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

- Local trend modeling via (Laplacian) temporal regularization:

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

- Property with discrete Fourier transform (denoted by $\mathcal{F}(\cdot)$)[2]:

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

[2] It refers to the Convolution theorem.

Circulant Matrix Nuclear Norm Minimization

Circulant Matrix Nuclear Norm Minimization (CircNNM)

For any partially observed time series $\boldsymbol{y} \in \mathbb{R}^T$ with observed index set Ω, the optimization problem of CircNNM for reconstructing time series is given by

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

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

- An important property:
  $$\|\mathcal{C}(\boldsymbol{x})\|_* = \|\mathcal{F}(\boldsymbol{x})\|_1$$
- CircNNM shows an efficient FFT[3] implementation in $\mathcal{O}(T \log T)$ time (Liu'22, Liu & Zhang'23).

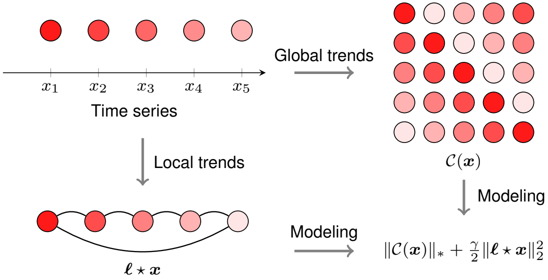
[3] Fast Fourier Transform (FFT).

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.,

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



Laplacian Convolutional Representation

- LCR model:

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

- Augmented Lagrangian function:

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

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

Laplacian Convolutional Representation

- LCR model:

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

- Augmented Lagrangian function:

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

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

- The ADMM scheme:

$$\begin{cases} \boldsymbol{x} := \arg\min_{\boldsymbol{x}} \ \mathcal{L}(\boldsymbol{x}, \boldsymbol{z}, \boldsymbol{w}) \\ \boldsymbol{z} := \arg\min_{\boldsymbol{z}} \ \mathcal{L}(\boldsymbol{x}, \boldsymbol{z}, \boldsymbol{w}) \\ \quad = \frac{1}{\lambda + \eta}\mathcal{P}_\Omega(\lambda \boldsymbol{x} + \boldsymbol{w} + \eta \boldsymbol{y}) + \frac{1}{\lambda}\mathcal{P}_\Omega^\perp(\lambda \boldsymbol{x} + \boldsymbol{w}) \\ \boldsymbol{w} := \boldsymbol{w} + \lambda(\boldsymbol{x} - \boldsymbol{z}) \end{cases}$$

Laplacian Convolutional Representation

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

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

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).

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:

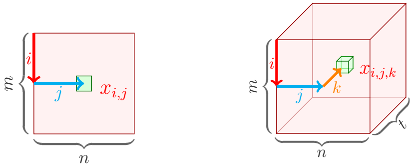
$$\min_{\hat{\boldsymbol{x}}} \ \|\hat{\boldsymbol{x}}\|_1 + \frac{\delta}{2}\|\hat{\boldsymbol{x}}-\hat{\boldsymbol{h}}\|_2^2$$

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

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

Laplacian Convolutional Representation

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



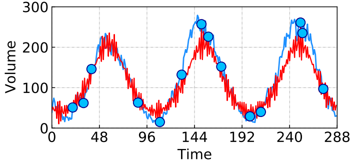
Two-Dimensional LCR (LCR-2D)

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

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

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

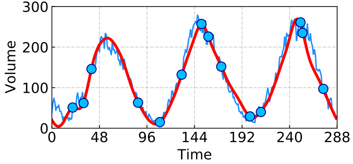
Experiments



CircNNM:

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

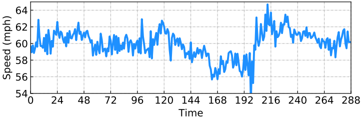
$\Downarrow$ Plus temporal regularization (TR)



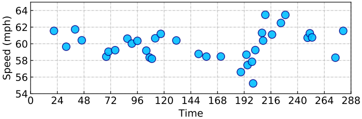
LCR:

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

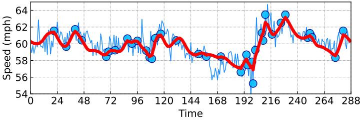
Experiments

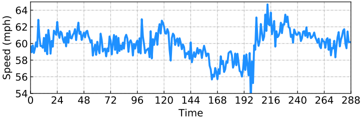


⇓ Mask **90%** observations

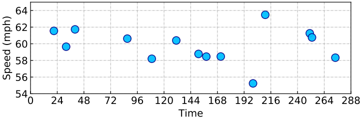


⇓ Reconstruct time series

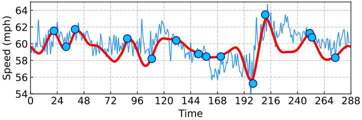




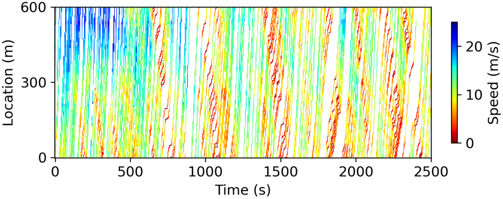
⇓ Mask **95%** observations



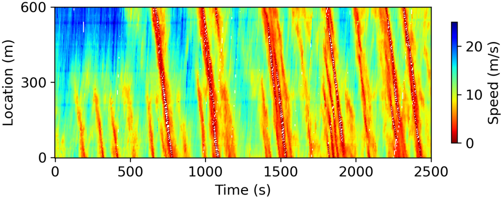
⇓ Reconstruct time series



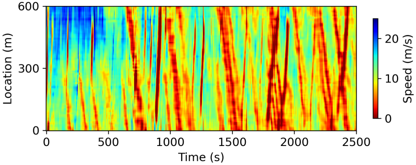
Experiments



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



Experiments

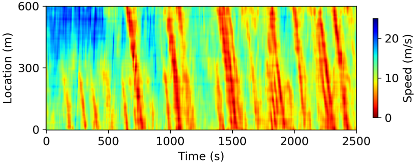


TR:

$$\min_{\boldsymbol{X}} \ \mathcal{R}_{\tau}(\boldsymbol{X})$$

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

$\Downarrow$ Plus CircNNM

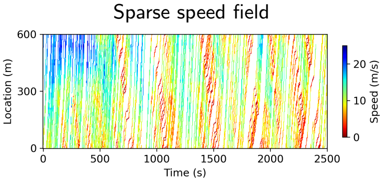


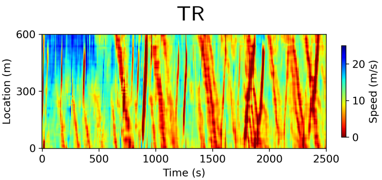
LCR:

$$\min_{\boldsymbol{X}} \ \|\mathcal{C}(\boldsymbol{X})\|_* + \gamma \cdot \mathcal{R}_{\tau}(\boldsymbol{X})$$

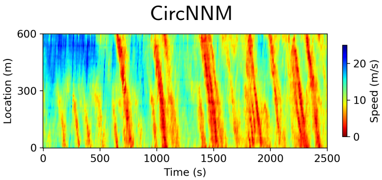
$$\text{s. t. } \ \|\mathcal{P}_{\Omega}(\boldsymbol{X}-\boldsymbol{Y})\|_F \leq \epsilon$$

Experiments

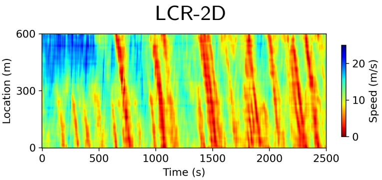




MAPE = 46.94% & RMSE = 4.34 m/s



MAPE = 43.51% & RMSE = 1.65 m/s



MAPE = **41.29%** & RMSE = **1.55 m/s**

Conclusion

- **(Starting point)** How to impute traffic time series?

  ✓ Low-rank models ✓ Temporal regularization

- **(Solution)** Time series trend modeling in the low-rank framework?
  - Global time series trend modeling (low-rank model):

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

  - Local time series trend modeling (temporal regularization):

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

- **(Highlight)** A unified framework with the FFT implementation.

References

A short list:

- **[Liu'22]** G. Liu (2022). Time series forecasting via learning convolutionally low-rank models. IEEE Transactions on Information Theory, 68(5): 3362–3380.
- **[Liu & Zhang'23]** G. Liu and W. Zhang (2023). Recovery of future data via convolution nuclear norm minimization. IEEE Transactions on Information Theory, 69(1): 650–665.

Thanks for your attention!

Any Questions?

About me:

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