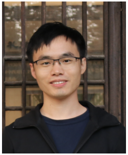
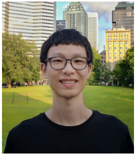
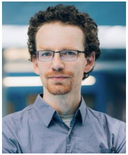
POLYTECHNIQUE MONTRÉAL
UNIVERSITÉ D'INGÉNIERIE

IVADO

Laplacian Convolutional Representation for Traffic Time Series Imputation

Xinyu Chen

July 19, 2023


Ph.D. Candidate	**Postdoc**	**Supervisor**	**Co-supervisor**
Xinyu Chen	Dr. Zhanhong Cheng	Prof. Nicolas Saunier	Prof. Lijun Sun
Polytechnique Montréal	McGill University	Polytechnique Montréal	McGill University

Current work:

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

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:

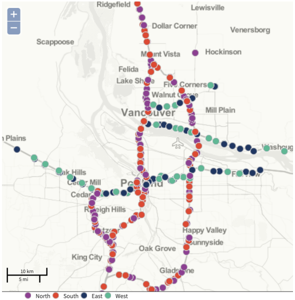
- https://xinychen.github.io/slides/LCR.pdf

Outline

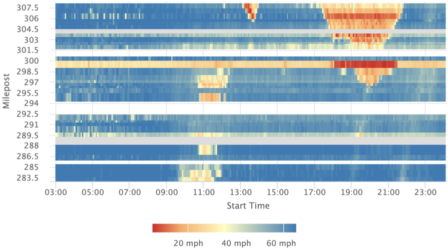
- **Motivation**
  - Data-Driven ITS
  - Time Series Imputation
  - Speed Field Reconstruction
- **Revisit Laplacian Matrix & Circular Convolution**
  - Laplacian Matrix
  - Laplacian Regularization
- **Circulant Matrix Nuclear Norm Minimization**
- **Laplacian Convolutional Representation**
  - Model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

Motivation

- Portland highway traffic flow data[1]



Highway network & sensor locations



Traffic speed field

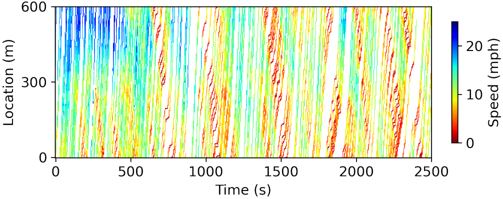
- Speed field $\boldsymbol{Y} \in \mathbb{R}^{N \times T}$ (N locations & T time steps)
- Speed field shows strong spatial/temporal dependencies

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

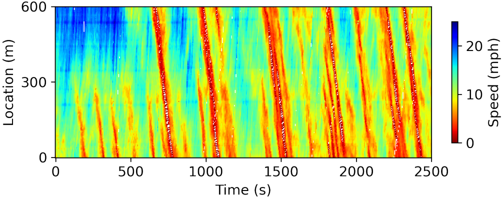
Motivation

- How to reconstruct missing values from partial observations?

Motivation



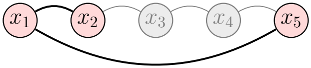
200-by-500 matrix (NGSIM) $\Downarrow$ Reconstruct speed field from 20% sparse trajectories?



- 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

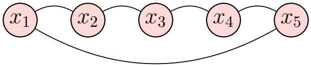
$\xrightarrow{\text{Modeling}}$

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

(Circulant) Laplacian matrix

Revisit Laplacian Matrix & Circular Convolution

- Intuition of (circulant) Laplacian matrix.



Undirected and circulant graph

$$\xrightarrow{\text{Modeling}}$$

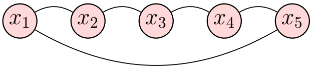
$$\boldsymbol{L} = \begin{bmatrix} \mathbf{2} & -1 & 0 & 0 & -1 \\ \mathbf{-1} & 2 & -1 & 0 & 0 \\ 0 & -1 & 2 & -1 & 0 \\ 0 & 0 & -1 & 2 & -1 \\ \mathbf{-1} & 0 & 0 & -1 & 2 \end{bmatrix}$$

(Circulant) Laplacian matrix

Revisit Laplacian Matrix & Circular Convolution

Reformulate Laplacian regularization with circular convolution.

- 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

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

- 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 $\mathcal{F}(\cdot)$):

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

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.

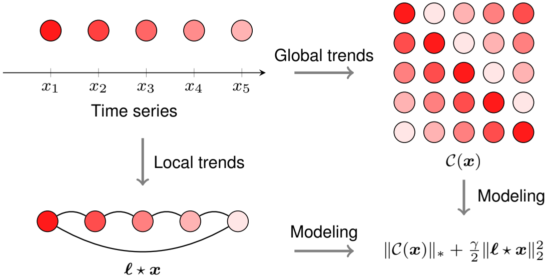
- An important property:
$$\|\mathcal{C}(\boldsymbol{x})\|_* = \|\mathcal{F}(\boldsymbol{x})\|_1$$
- CircNNM shows an efficient FFT implementation (Liu'22).

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.

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

$$\begin{aligned} \boldsymbol{x} &:= \arg\min_{\boldsymbol{x}}\ \|\mathcal{C}(\boldsymbol{x})\|_* + \frac{\gamma}{2}\|\boldsymbol{\ell}\star\boldsymbol{x}\|_2^2 + \frac{\lambda}{2}\|\boldsymbol{x}-\boldsymbol{z}+\boldsymbol{w}/\lambda\|_2^2 \\ \Longrightarrow \hat{\boldsymbol{x}} &:= \arg\min_{\hat{\boldsymbol{x}}}\ \|\hat{\boldsymbol{x}}\|_1 + \frac{\gamma}{2T}\|\hat{\boldsymbol{\ell}}\circ\hat{\boldsymbol{x}}\|_2^2 + \frac{\lambda}{2T}\|\hat{\boldsymbol{x}}-\hat{\boldsymbol{z}}+\hat{\boldsymbol{w}}/\lambda\|_2^2 \\ &= \arg\min_{\hat{\boldsymbol{x}}}\ \|\hat{\boldsymbol{x}}\|_1 + \frac{\lambda}{2T}\|\hat{\boldsymbol{x}}-(\lambda\hat{\boldsymbol{z}}-\hat{\boldsymbol{w}})\oslash(\lambda\hat{\boldsymbol{\ell}}^*\circ\hat{\boldsymbol{\ell}}+\lambda\mathbb{1}_T)\|_2^2 \end{aligned}$$

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

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

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

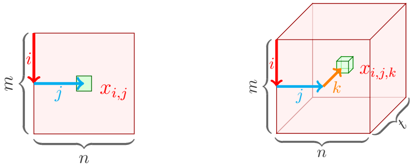
$$\min_{\hat{\boldsymbol{x}}}\ \|\hat{\boldsymbol{x}}\|_1 + \frac{\lambda}{2T}\|\hat{\boldsymbol{x}}-\hat{\boldsymbol{h}}\|_2^2$$

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

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

Laplacian Convolutional Representation

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



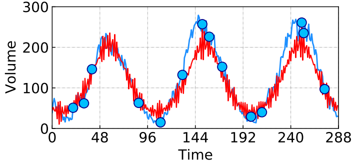
Two-Dimensional LCR (LCR-2D)

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.

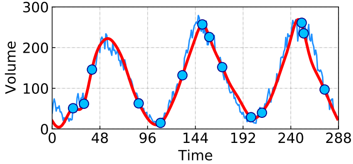
Experiments



CircNNM:

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

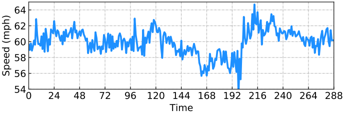
$\Downarrow$ Plus temporal regularization



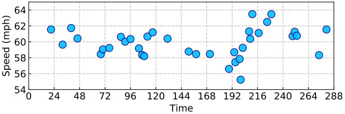
LCR:

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

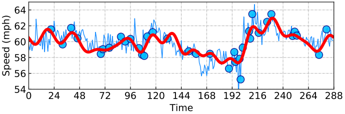
Experiments

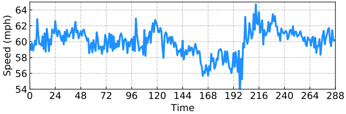


Mask **90%** observations

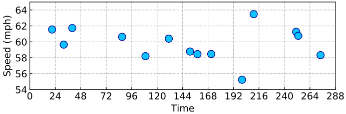


Reconstruct time series

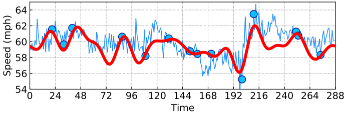




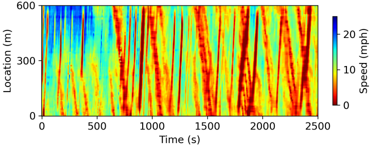
Mask **95%** observations



Reconstruct time series

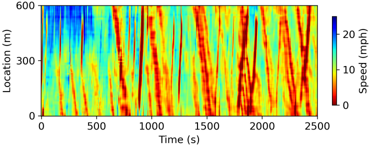


Quadratic variation completion (QVC)



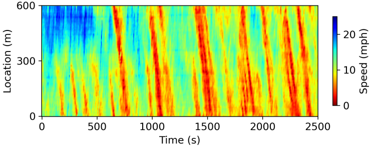
MAPE = 51.50% & RMSE = 4.86mph

Laplacian kernelized completion (LKC)



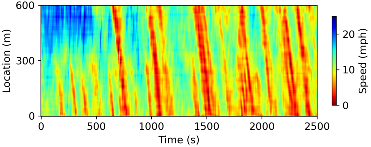
MAPE = 46.94% & RMSE = 4.34mph

Circulant tensor nuclear norm minimization (CTNNM)



MAPE = 43.51% & RMSE = 1.65mph

Laplacian convolutional representation (LCR)



MAPE = **41.29%** & RMSE = **1.55mph**

- QVC & LKC:

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

- CTNNM:

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

Conclusion

- **(Starting point)** How to reconstruct sparse speed field?
  ✓ Matrix factorization (**MF**) ✓ Tensor factorization (**TF**)
- **(Highlight)** The importance of spatiotemporal modeling in low-rank methods?
  - Spatial/temporal **smoothing** regularization:

$$\min_{\boldsymbol{W},\boldsymbol{X}} \frac{1}{2}\left\|\mathcal{P}_{\Omega}(\boldsymbol{Y}-\boldsymbol{W}^\top\boldsymbol{X})\right\|_F^2+\frac{\rho}{2}(\|\boldsymbol{W}\|_F^2+\|\boldsymbol{X}\|_F^2)$$
$$+\frac{\lambda}{2}(\|\boldsymbol{W}\boldsymbol{\Psi}_1^\top\|_F^2+\|\boldsymbol{X}\boldsymbol{\Psi}_2^\top\|_F^2)$$

  - Automatic temporal modeling via **Hankelization**:

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

vs.

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

- **(Challenge)** How to solve the high memory consumption issue in the Hankelization process?

References

A short list:

- **[Liu'22]** x
- **[Liu & Zhang'22]** x

Thanks for your attention!

Any Questions?

About me:

- Homepage: https://xinychen.github.io
- GitHub: https://github.com/xinychen (**3.4k+ stars**)
- Blog: https://medium.com/@xinyu.chen (**70k+ views**)
- How to reach me: chenxy346@gmail.com