Modeling Temporal Correlations and Dynamics in Spatiotemporal Data Systems

Xinyu Chen

April 19, 2024

Outline

A quick look:

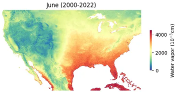
- Motivation
- Autoregression on spatiotemporal systems
- Tensor factorization
- Dynamic autoregressive tensor factorization
- Benchmark evaluation
- International trade
- Human mobility
- Applications to M3S
- Brainstorming

Motivation

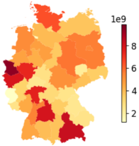
- Spatiotemporal systems & data scenarios

Transportation

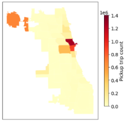
Mobile service



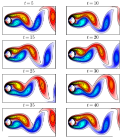
Climate



Energy



Mobility



Fluid flow

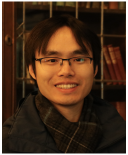
- Prior art: e.g., dynamic mode decomposition, matrix/tensor factorization
- Challenges: Time-varying system, multidimensional system (e.g., human mobility)

Prior Art

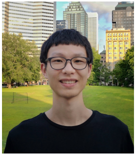
- Autoregression, attention-based sequence models
- Machine learning tasks: estimation, imputation/interpolation, prediction

Laplacian Convolutional Representation for Traffic Time Series Imputation

1st round revision at
IEEE Transactions on Knowledge and Data Engineering

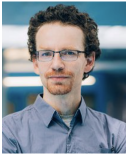


Dr. Xinyu Chen



Dr. Zhanhong Cheng

Prof. HanQin Cai



Prof. Nicolas Saunier

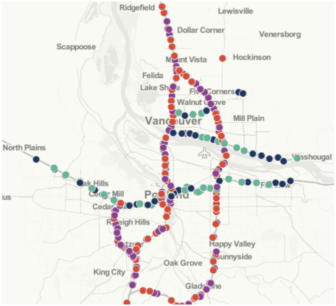
Prof. Lijun Sun

Materials:

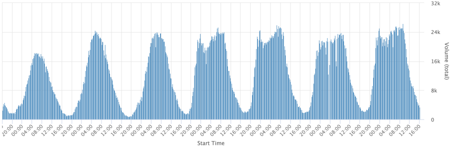
- PDF: https://xinychen.github.io/papers/Laplacian_convolution.pdf
- GitHub: https://github.com/xinychen/transdim (1.1k+ stars)

Traffic Flow Data

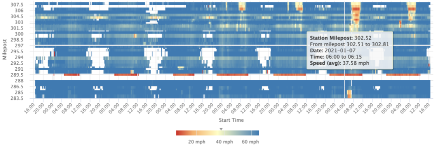
- Portland highway traffic data[1]



Highway network & sensor locations



Traffic volume



Traffic speed

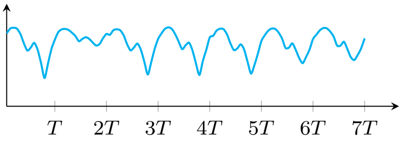
- $\boldsymbol{X} \in \mathbb{R}^{N \times T}$ with N spatial locations $\times$ T time steps
- Traffic volume/speed shows strong spatial/temporal dependencies

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

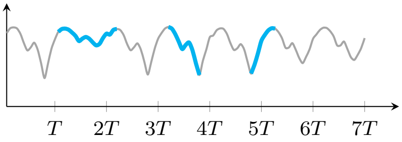
Time Series Imputation

Motivation: Traffic imputation

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



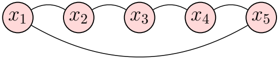
- Local trends (e.g., short-term time series trends):



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

Local Trend Modeling

- Intuition of (circulant) Laplacian matrix



Undirected and circulant graph

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

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

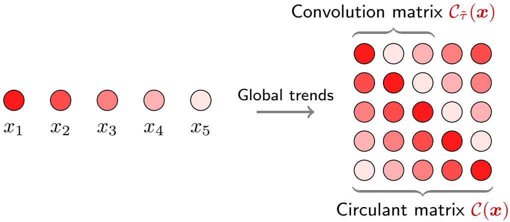
- (Laplacian) Temporal regularization:

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

Reformulate temporal regularization with circular convolution.

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.

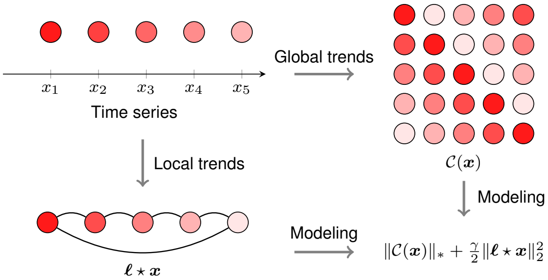
Global + Local Trends?

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.,

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

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



Laplacian Convolutional Representation

- Augmented Lagrangian function:[2]

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

- The ADMM scheme:

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

- 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.

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

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:

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

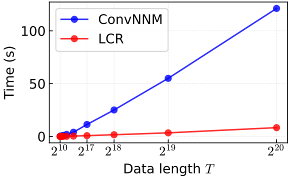
with complex-valued $\hat{\boldsymbol{x}},\hat{\boldsymbol{h}}\in\mathbb{C}^T$ and weight parameter δ, element-wise, the solution is given by

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

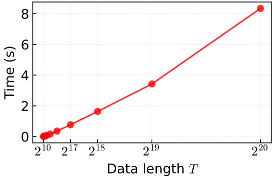
Empirical time complexity

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** (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 $\mathcal{O}(\tilde{\tau}^2 T)$

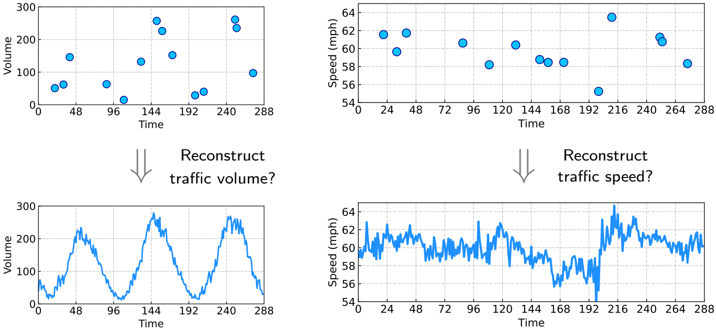


ConvNNM vs. LCR



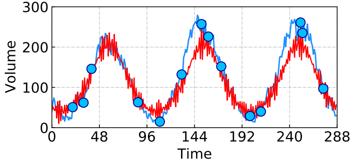
LCR

Experiments



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

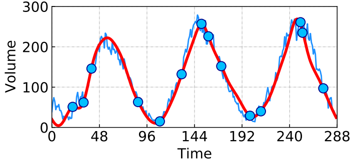
Experiments



CircNNM:

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

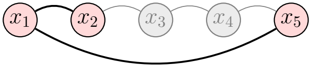
$\Downarrow$ Plus temporal regularization (TR)



LCR:

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

- The start data points and end data points are connected?



Undirected and circulant graph

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

(Circulant) Laplacian matrix

- Flipping operation on $\boldsymbol{x} \in \mathbb{R}^5$:

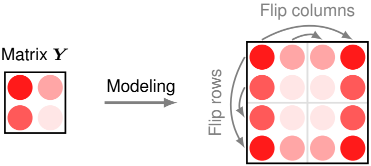
$$\boldsymbol{x}_{\text{new}} = \begin{bmatrix} \boldsymbol{x} \\ \boldsymbol{J}\boldsymbol{x} \end{bmatrix} = (\underbrace{x_1, x_2, x_3, x_4, x_5,}_{\text{Original time series}} \quad \underbrace{x_5, x_4, x_3, x_2, x_1}_{\text{Flipped time series}})^\top \in \mathbb{R}^{10}$$

  where $\boldsymbol{J} \in \mathbb{R}^{5 \times 5}$ is the exchange matrix.

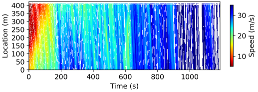
- Potential applications: Passenger flow prediction with strong global/local trends

Speed field reconstruction[3]

- Flipping operation on a matrix:

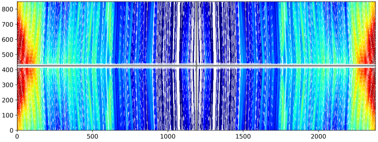


- Flipping operation on a speed field of vehicular traffic flow:



Original speed field

Modeling



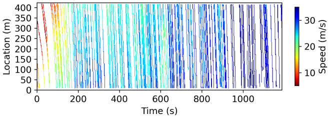
Flipped speed field

[3] Highway Drone (HighD) dataset at https://www.highd-dataset.com/

Speed field reconstruction[4]

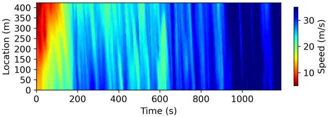
- Scenario: Mask trajectories of 70% vehicles
- LCR-2D on partially observed $\boldsymbol{Y} \in \mathbb{R}^{N \times T}$:

$$\min_{\boldsymbol{X}} \quad \underbrace{\|\mathcal{C}(\boldsymbol{X})\|_*}_{\text{Global trend}} + \frac{\gamma}{2} \underbrace{\|(\boldsymbol{\ell}_s \boldsymbol{\ell}^\top) \star \boldsymbol{X}\|_F^2}_{\text{Local trend}}$$
$$\text{s.t. } \|\mathcal{P}_\Omega(\boldsymbol{X} - \boldsymbol{Y})\|_F \leq \epsilon$$

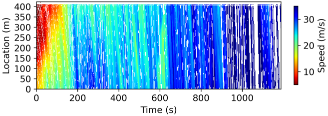


Sparse speed field

LCR-2D →



Reconstructed speed field



Ground-truth speed field

[4] Highway Drone (HighD) dataset at https://www.highd-dataset.com/

Contributions

Matrix nuclear norm ($\|\boldsymbol{X}\|_*$) minimization

Candès & Recht'09

Singular value thresholding

Cai et al.'10

Truncated nuclear norm ($\|\boldsymbol{X}\|_{r,*}, r \in \mathbb{Z}^+$) 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

(Ours) **LCR**:
- ✓ Local trend modeling
- ✓ An FFT implementation

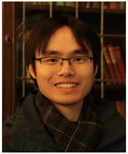
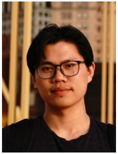
Vision & Insight

Rethinking the importance of local trend modeling in traffic data imputation tasks.
Finding a unified global and local trend modeling framework whose optimization can efficiently solved by FFT with log-linear time complexity.
Reduce model biases? Difference between imputation and interpolation?
Estimating network-wide traffic states? Planning & management

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

IEEE Transactions on Knowledge and Data Engineering, 2024

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

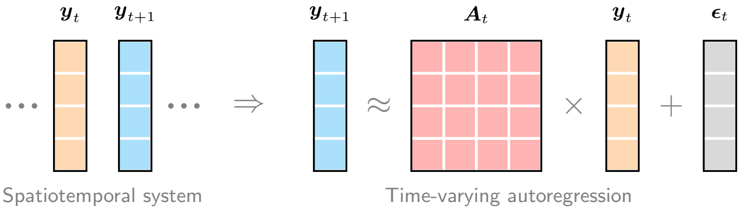

Dr. Xinyu Chen | Chengyuan Zhang⋆ | Xiaoxu Chen | Prof. Nicolas Saunier | Prof. Lijun Sun

Materials:

- PDF: https://xinychen.github.io/papers/time_varying_model.pdf
- GitHub: https://github.com/xinychen/vars

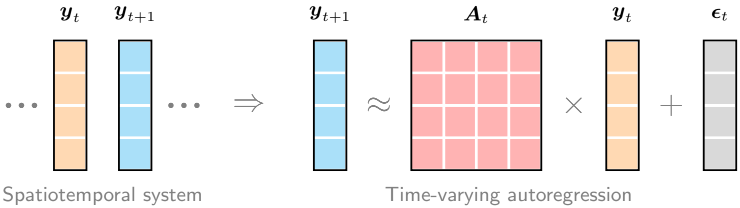
Autoregression

- How to characterize dynamical systems?



Autoregression

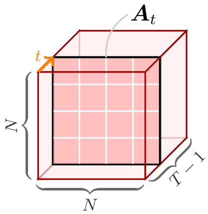
- How to characterize dynamical systems?

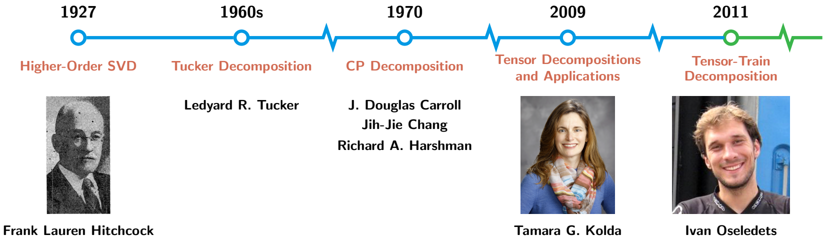


- On spatiotemporal systems $\boldsymbol{Y} \in \mathbb{R}^{N \times T}$:

$$\underbrace{\boldsymbol{y}_{t+1} = \boldsymbol{A}\boldsymbol{y}_t + \boldsymbol{\epsilon}_t}_{\text{Time-invariant}} \quad \text{v.s.} \quad \underbrace{\boldsymbol{y}_{t+1} = \boldsymbol{A}_t\boldsymbol{y}_t + \boldsymbol{\epsilon}_t}_{\text{Time-varying}}$$

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




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

Time-Varying Autoregression

xxxx

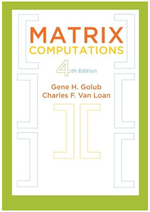
DATF

- Tensor factorization[5]:

$$\mathcal{A} = \underbrace{\mathcal{G} \times_1 \boldsymbol{W} \times_2 \boldsymbol{V} \times_3 \boldsymbol{X}}_{\text{Tucker decomposition}}$$

$$\Updownarrow$$

$$\boldsymbol{A}_t = \mathcal{G} \times_1 \underbrace{\boldsymbol{W}}_{\text{spatial modes}} \times_2 \boldsymbol{V} \times_3 \underbrace{\boldsymbol{x}_t^\top}_{\text{temporal modes}}$$



- **(Ours)** Dynamic autoregressive tensor factorization (DATF):

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

$$\text{s.t.} \quad \underbrace{\boldsymbol{W}^\top \boldsymbol{W} = \boldsymbol{I}_R}_{\text{orthogonal spatial modes}}$$

- Solution: $\mathcal{G}$ (LS) $\rightarrow$ $\boldsymbol{W}$ **(OPP)** $\rightarrow$ $\boldsymbol{V}$ (CG) $\rightarrow$ $\boldsymbol{x}_t$ (LS)

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

OPP

- **Orthogonal Procrustes problem**: For any $\boldsymbol{Q} \in \mathbb{R}^{m\times r}$, $m \geq r$, the solution to

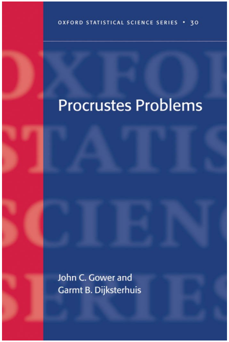
$$\min_{\boldsymbol{F}} \ \|\boldsymbol{F} - \boldsymbol{Q}\|_F^2 \quad \text{s.t.} \ \underbrace{\boldsymbol{F}^\top \boldsymbol{F} = \boldsymbol{I}_r}_{\text{orthogonal}}$$

is

$$\boldsymbol{F} := \boldsymbol{U}\boldsymbol{V}^\top$$

where

$$\underbrace{\boldsymbol{Q} = \boldsymbol{U}\boldsymbol{\Sigma}\boldsymbol{V}^\top}_{\text{singular value decomposition}}$$



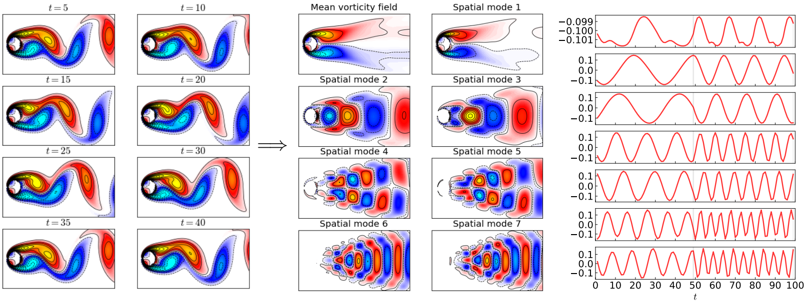
- Equivalent form:

$$\|\boldsymbol{F} - \boldsymbol{Q}\|_F^2 = \mathrm{tr}(\underbrace{\boldsymbol{F}^\top \boldsymbol{F}}_{=\boldsymbol{I}_r} - \boldsymbol{F}^\top \boldsymbol{Q} - \boldsymbol{Q}^\top \boldsymbol{F} + \underbrace{\boldsymbol{Q}^\top \boldsymbol{Q}}_{\text{const.}}) = -2\,\mathrm{tr}(\boldsymbol{F}^\top \boldsymbol{Q}) + \text{const.}$$

$$\Longrightarrow \boldsymbol{F} =: \mathop{\arg\min}_{\boldsymbol{F}^\top \boldsymbol{F} = \boldsymbol{I}_r} \ \|\boldsymbol{F} - \boldsymbol{Q}\|_F^2 = \mathop{\arg\max}_{\boldsymbol{F}^\top \boldsymbol{F} = \boldsymbol{I}_r} \ \mathrm{tr}(\boldsymbol{F}^\top \boldsymbol{Q})$$

Benchmark Evaluation

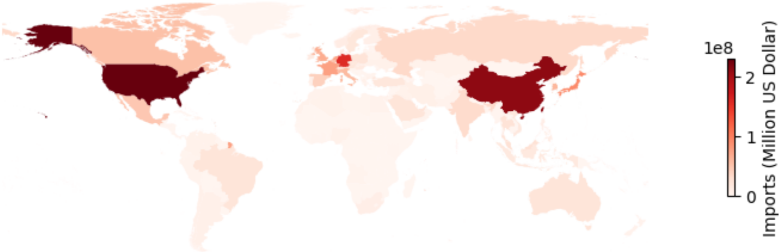
- **Multi-resolution fluid flow dataset** (the first 50 snapshots + 50 snapshots randomly selected from the last 100 snapshots)
  - Produce interpretable patterns: Low-frequency modes (dominant patterns) & high-frequency modes (e.g., secondary patterns, outliers)
  - Identify the system of different frequencies (i.e., at $t = 50$)



Vectorize snapshots as $\{\boldsymbol{y}_t\}_t$ Spatial modes in $\boldsymbol{W}$ Temporal modes in $\boldsymbol{X}$

International Trade

- **Merchandise import trade values (annual)**[6] (222 countries/regions & period of 2000-2022)
  - Aggregate over 18 product types (e.g., Agricultural products, Food, Manufactures, Chemicals, etc.)
  - Represent such data as a 222-by-22 matrix



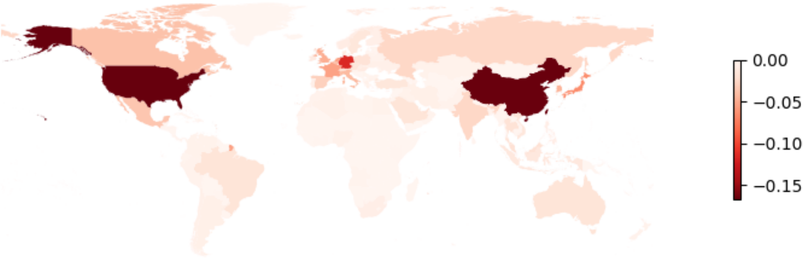
Aggregated import trade values from 2000 to 2022

[6] The dataset is available at https://www.wto.org/english/res_e/statis_e/trade_datasets_e.htm.

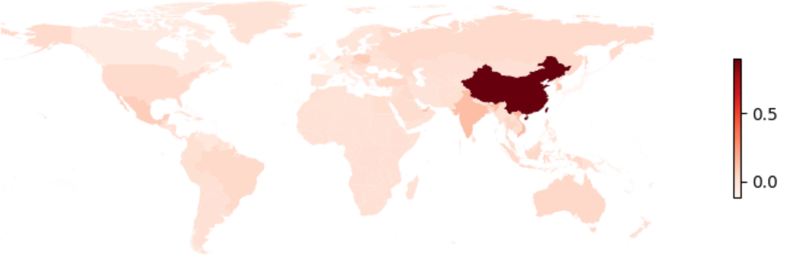
International Trade

Product/Sector types:

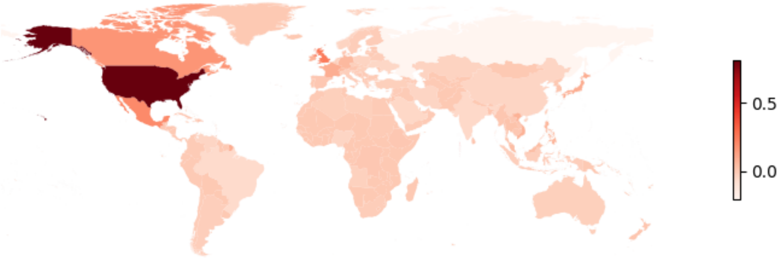
-



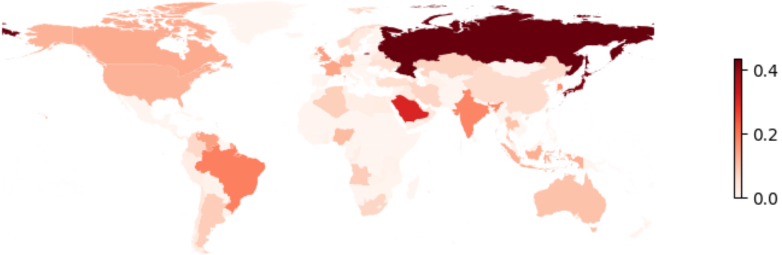
The **1st spatial mode** in the matrix $\boldsymbol{W}$.



The **2nd spatial mode** in the matrix $\boldsymbol{W}$.

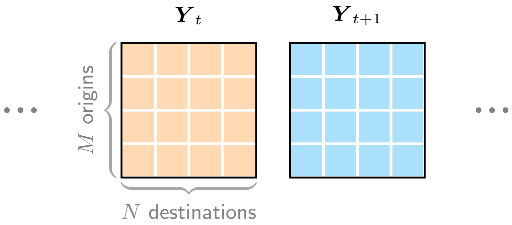


The **3rd spatial mode** in the matrix $\boldsymbol{W}$.



The **4th spatial mode** in the matrix $\boldsymbol{W}$.

- Origin-Destination (OD) matrices



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

$$\underbrace{\boldsymbol{y}_{n,t+1} = \boldsymbol{A}_{n,t} \boldsymbol{y}_{n,t} + \boldsymbol{\epsilon}_{n,t}}_{\text{time-varying \& destination-varying}}$$

- Optimization problem of DATF:

$$\begin{aligned} \min_{\boldsymbol{\mathcal{G}}, \boldsymbol{W}, \boldsymbol{U}, \boldsymbol{V}, \boldsymbol{X}} \quad & \frac{1}{2} \sum_{n \in [N]} \sum_{t \in [T-1]} \left\| \boldsymbol{y}_{n,t+1} - (\boldsymbol{\mathcal{G}} \times_1 \boldsymbol{W} \times_2 \boldsymbol{U} \times_3 \boldsymbol{V} \times_4 \boldsymbol{x}_t^\top) \boldsymbol{y}_{n,t} \right\|_2^2 \\ \text{s.t.} \quad & \underbrace{\boldsymbol{W}^\top \boldsymbol{W} = \boldsymbol{I}_R}_{\text{orthogonal origin modes}} \end{aligned}$$

Human Mobility

- Chicago taxi/ridesharing data

Matching Taxi Trips with Community Areas

There are three basic steps to follow for processing taxi trip data:

- Download taxi trips in 2022 in the `.csv` format, e.g., `Taxi_Trips_-_2022.csv`.
- Use the `pandas` package in Python to process the raw trip data.
- Match trip pickup/dropoff locations with boundaries of the community area.

```
import pandas as pd

data = pd.read_csv('Taxi_Trips_-_2022.csv')
data.head()
```

For each taxi trip, one can select some important information:

- `Trip Start Timestamp`: When the trip started, rounded to the nearest 15 minutes.
- `Trip Seconds`: Time of the trip in seconds.
- `Trip Miles`: Distance of the trip in miles.
- `Pickup Community Area`: The Community Area where the trip began. This column will be blank for locations ourside Chicago.
- `Dropoff Community Area`: The Community Area where the trip ended. This column will be blank for locations outside Chicago.

```
df = pd.DataFrame()
df['Trip Start Timestamp'] = data['Trip Start Timestamp']
df['Trip Seconds'] = data['Trip Seconds']
df['Trip Miles'] = data['Trip Miles']
df['Pickup Community Area'] = data['Pickup Community Area']
df['Dropoff Community Area'] = data['Dropoff Community Area']
del data
df
```

Figure 2 shows taxi pickup and dropoff trips (2022) on 77 community areas in the City of Chicago. Note that the average trip duration is **1207.75 seconds** and the average trip distance is **6.16 miles**.

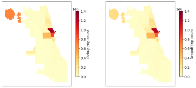


Figure 2. Taxi pickup and dropoff trips (2022) in the City of Chicago, USA. There are 4,763,961 remaining trips after the data processing.

For comparison, Figure 3 shows taxi pickup and dropoff trips (2019) on 77 community areas in the City of Chicago. Note that the average trip duration is **915.62 seconds** and the average trip distance is **3.93 miles**.

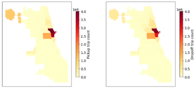


Figure 3. Taxi pickup and dropoff trips (2019) in the City of Chicago, USA. There are 12,484,572 remaining trips after the data processing. See the data processing codes.

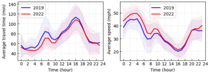


Figure 6. Average travel time and speed from area 32 (i.e., Downtown) to area 76 (i.e., Airport) in both 2019 and 2022.

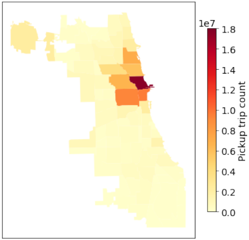
```
import numpy as np
import matplotlib.pyplot as plt

fig = plt.figure(figsize = (8, 2.5))
ax = fig.add_subplot(1, 2, 1)
# Average travel time in 2019
m1 = df1.groupby(['hour'])['Trip Seconds'].mean().values / 30
s1 = df1.groupby(['hour'])['Trip Seconds'].std().values / 30
plt.plot(m1, color = 'blue', linewidth = 1.8, label = '2019')
upper = m1 + s1
lower = m1 - s1
x_bound = np.append(np.append(np.append(np.array([0, 0]), np.arange(0, 24)),
                    np.array([24 - 1, 24 - 1])), np.arange(24 - 1, -1,
y_bound = np.append(np.append(np.append(np.array([upper[0], lower[0]]), lower),
                    np.array([lower[-1], upper[-1]])), np.flip(upper))
plt.fill(x_bound, y_bound, color = 'blue', alpha = 0.05)

# Average travel time in 2022
m1 = df2.groupby(['hour'])['Trip Seconds'].mean().values / 30
s1 = df2.groupby(['hour'])['Trip Seconds'].std().values / 30
plt.plot(m1, color = 'red', linewidth = 1.8, label = '2022')
upper = m1 + s1
lower = m1 - s1
```

Source: https://spatiotemporal-data.github.io/Chicago-mobility/taxi-data

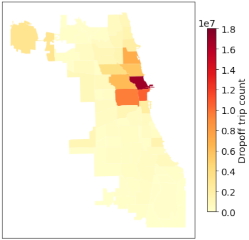
Human Mobility

- **Ridesharing:** 96,642,881 trips in 2019 vs. 57,290,954 trips in 2022

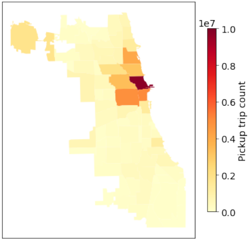
Pickup trips (2019)



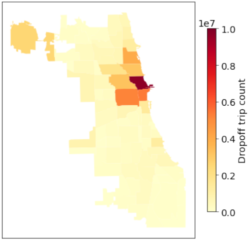
Dropoff trips (2019)



Pickup trips (2022)



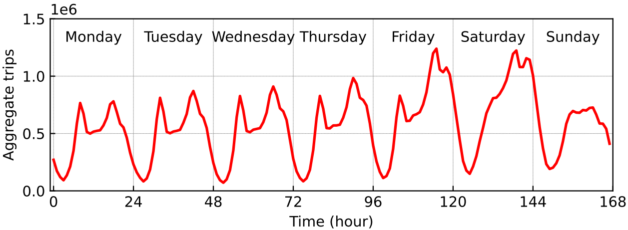
Dropoff trips (2022)



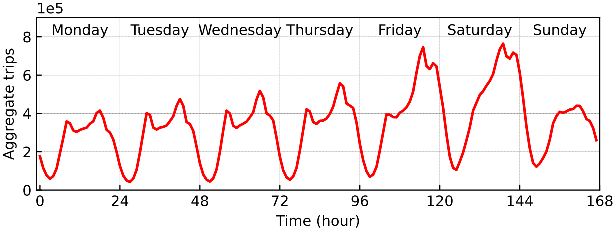
Human Mobility

- **Ridesharing:** 96,642,881 trips in 2019 vs. 57,290,954 trips in 2022

Pickup trips aggregated over 52 weeks in 2019

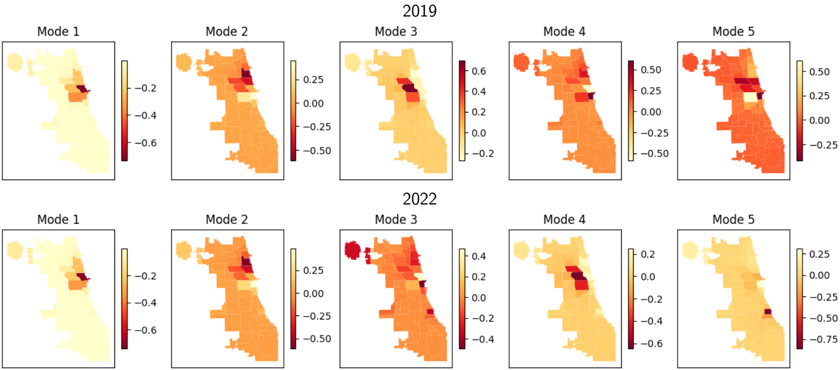


Pickup trips aggregated over 52 weeks in 2022



Human Mobility

- Ridesharing trip data: 77 origins $\times$ 77 destinations $\times$ 168 hours
- Our model Identifies the changes in pickup zones before and after COVID-19



Application to M3S

Main tasks:

- Pre-process the urban datasets (e.g., Veraset) in Singapore[7]
- Define the scientific questions in the project
- Formulate the problems with machine learning
- Analyze the results and their impacts

Current ideas: Discovering dynamics of urban human activity with dynamic autoregressive tensor factorization

- (On 2D activity data) Uncover spatial modes/patterns (e.g., POI patterns)
- (On 3D mobility data) Uncover temporal modes/patterns (e.g., long-term changing behavior impacted by special events and policy)

[7] https://spatiotemporal-data.github.io/trajectory/veraset/

Brainstorming: Put Some New Ideas

Basic assumption: time-varying systems, information compression;
Applications: pattern discovery, anomaly detection (in high frequency), and prediction
Data-driven urban planning:

- Shift of land-use in the past decades (hope to see the land-use patterns that evolve over time)
  - gradually changed over time
  - a kind of random process
  - involve causal factors (evaluate the inference capability)
- Connect the long-term mobility change (e.g., multiple travel modes) with land-use in the framework (which is not only the autoregression, may need a causal inference framework)

Future urban systems (should not only stay with basic concepts and use cases!):

- Rethink the electric vehicle charging stations from the land-use perspective against fuel stations
- How to characterize the relationship between mobility transition and sustainability

Brainstorming: Put Some New Ideas

People's career trajectory:

- Introduce abstract notion spaces (e.g., work type, time resolution) for detecting long-term changes in individual career
- dimensions such as (people, education stage)

International trade:

- How to represent (import, export) networks?
- How to discover spatial patterns from three-dimensional or even higher-dimensional trade data, e.g., on dimensions (country/region, product type, year)?
- Reference: The building blocks of economic complexity (Hidalgo & Hausmann'09 at PNAS)

Thanks for your attention!

Any Questions?

https://xinychen.github.io/slides/temporal_modeling.pdf

About me:

- Homepage: https://xinychen.github.io
- How to reach me: chenxy346@gmail.com
- Or through: xinychen@mit.edu