Research Interests & Past Works

Xinyu Chen

Postdoc, MIT, USA

Ph.D., University of Montreal, Canada

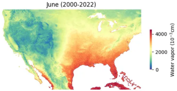
March 19, 2023

Research Interests

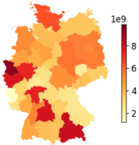
- Machine Learning (e.g., Matrix/Tensor computations, numerical optimization, supervised/unsupervised learning)
- Data Mining (e.g., Spatiotemporal data modeling, geospatial data analysis)

Transportation

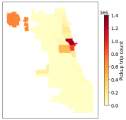
Mobile service



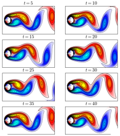
Climate



Energy



Mobility



Dynamical system

- AI for Science (e.g., Urban science, dynamical systems, computational social science)

Past Works

Spatiotemporal traffic data imputation:

1. X. Chen, Z. He, J. Wang (2018). Spatial-temporal traffic speed patterns discovery and incomplete data recovery via SVD-combined tensor decomposition. *Transportation Research Part C: Emerging Technologies*. 86: 59-77. (100+ citations)
2. X. Chen, Z. He, L. Sun (2019). A Bayesian tensor decomposition approach for spatiotemporal traffic data imputation. *Transportation Research Part C: Emerging Technologies*. 98: 73-84. (200+ citations, **ESI highly cited paper**)
3. X. Chen, Z. He, Y. Chen, Y. Lu, J. Wang (2019). Missing traffic data imputation and pattern discovery with a Bayesian augmented tensor factorization model. *Transportation Research Part C: Emerging Technologies*. 104: 66-77. (100+ citations)
4. X. Chen, J. Yang, L. Sun (2020). A nonconvex low-rank tensor completion model for spatiotemporal traffic data imputation. *Transportation Research Part C: Emerging Technologies*. 117: 102673. (100+ citations)
5. X. Chen, Y. Chen, N. Saunier, L. Sun (2021). Scalable low-rank tensor learning for spatiotemporal traffic data imputation. *Transportation Research Part C: Emerging Technologies*. 129: 103226.

Past Works

Spatiotemporal traffic data imputation:

6. X. Chen, M. Lei, N. Saunier, L. Sun (2022). Low-rank autoregressive tensor completion for spatiotemporal traffic data imputation. *IEEE Transactions on Intelligent Transportation Systems*. 23 (8): 12301-12310. (50+ citations, **ESI hot paper**)

Spatiotemporal time series forecasting:

7. X. Chen, L. Sun (2022). Bayesian temporal factorization for multidimensional time series prediction. *IEEE Transactions on Pattern Analysis and Machine Intelligence*. 44 (9): 4659-4673. (150+ citations, **ESI hot paper** & **ESI highly cited paper**)

- X. Chen, X.L. Zhao, C. Cheng (2024). Forecasting urban traffic states with sparse data using Hankel temporal matrix factorization. INFORMS Journal on Computing. (**UTD-24**, minor revision for final dicision)

Spatiotemporal pattern discovery:

8. X. Chen, C. Zhang, X. Chen, N. Saunier, L. Sun (2024). Discovering dynamic patterns from spatiotemporal data with time-varying low-rank autoregression. *IEEE Transactions on Knowledge and Data Engineering*. 36 (2): 504-517.

Past Works

Open-source & reproducible research:

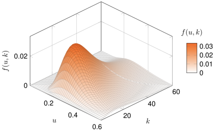
- https://github.com/xinychen

Algorithms

transdim

(1.1k stars)

Tools



awesome-latex-drawing

(1.2k stars)

Tutorials

latex-cookbook

(1.2k stars)

(THU Press)

Future Plan

Broad interests:

- Machine Learning
- Data Science/Mining
- AI for Science

Research directions:

- Urban science (e.g., connection among infrastructure, mobile/mobility activities, urban development, and economy)
- Human mobility modeling (e.g., long-range sequence prediction)
- Geospatial data analysis (e.g., orthogonal mode decomposition)
- Intelligent & sustainable urban systems
- Optimization & decision making
- ......

Goals: Solving many scientific, mathematical, and engineering problems with AI algorithms.

Spatiotemporal Data Modeling (STD) Initiative

- Homepage: https://spatiotemporal-data.github.io
- Areas: Data science & machine learning & AI for science
- What will we do?
  - Coding and computing with data
  - Posting scientific questions
  - Supporting open-source and reproducible research

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**.

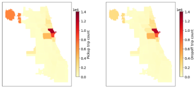


Figure 2. Taxi pickup and dropoff trips (2022) in the City of Chicago, USA. There are 4,763,961 remaining trips after the data processing.

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**.

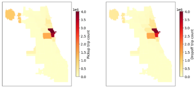


Figure 3. Taxi pickup and dropoff trips (2019) in the City of Chicago, USA. There are 12,484,572 remaining trips after the data processing. See the data processing codes.

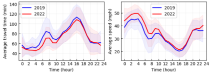


Figure 6. Average travel time and speed from area 32 (i.e., Downtown) to area 76 (i.e., Airport) in both 2019 and 2022.

```
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

Thanks for your attention!

Any Questions?

Slides: https://xinychen.github.io/slides/sustech24.pdf

About me:

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