

Spatiotemporal Data Visualization with Python

A Course for Undergraduate Students
@Southern University of Science and Technology

Xinyu Chen

March 31, 2023



南方科技大学



Southern University
of Science and
Technology

A Quick Look

Content

-

For your need!

- Slides: https://xinychen.github.io/slides/visual_stdta.pdf
- Examples: <https://spatiotemporal-data.github.io>

Target

Throughout this class, you will:

- Understand some basic concepts of spatiotemporal data
- Code and compute with spatiotemporal data in Python
- Visualize and analyze spatiotemporal data

Python

TBD

Geospatial Data

TBD

Research Interests

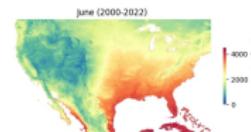
- **Machine Learning** (e.g., Matrix/Tensor computations, numerical optimization, supervised/unsupervised learning)
- **Data Mining** (e.g., Spatiotemporal data modeling, geospatial data analysis)



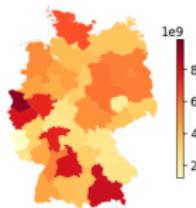
Transportation



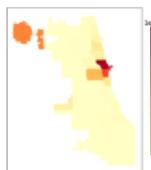
Mobile service



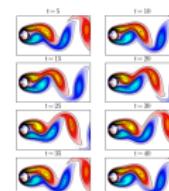
Climate



Energy



Mobility



Dynamical system

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

Past Works

Open-source & reproducible research:

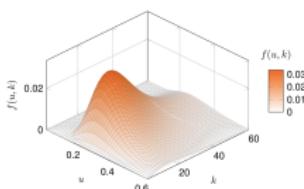
- <https://github.com/xinychen>

Algorithms



transdim
(1.1k stars)

Tools



awesome-latex-drawing
(1.2k stars)

Tutorials



latex-cookbook
(1.2k stars)
(THU Press)

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 outside 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
```

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 1207.76 seconds and the average trip distance is 6.18 miles.

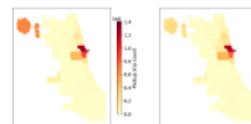


Figure 2. Taxi pickup and dropoff trips (2022) in the City of Chicago, USA. There are 4,763,961 remaining trips after the data processing.

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 5.93 miles.

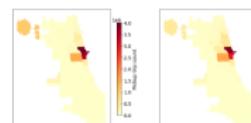


Figure 3. Taxi pickup and dropoff trips (2019) in the City of Chicago, USA. There are 12,484,572 remaining trips after the data processing. See the data processing codes.

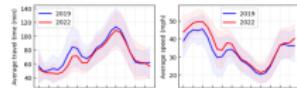


Figure 4. Average travel time and speed from area 32 (i.e., Downtown) to area 76 (i.e., Airport) in both 2019 and 2022.

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

fig = plt.figure(figsize=(4, 2.5))
ax = fig.add_subplot(1, 1, 1)
ax.set_title('Average Travel Time')

m1 = df1.groupby(['Hour'])[['Trip_Seconds']].mean() / 30
m1 = m1.rolling(12).mean().dropna()
lower, upper = m1 - 1.96 * m1.rolling(12).std(), m1 + 1.96 * m1.rolling(12).std()
ax.fill_between(lower.index, lower.values, upper.values, color='blue', alpha=0.5)

m2 = df2.groupby(['Hour'])[['Trip_Seconds']].mean() / 30
m2 = m2.rolling(12).mean().dropna()
lower, upper = m2 - 1.96 * m2.rolling(12).std(), m2 + 1.96 * m2.rolling(12).std()
ax.fill_between(lower.index, lower.values, upper.values, color='red', alpha=0.5)

ax.set_xlabel('Time (Hour)')
ax.set_ylabel('Average Travel Time (sec)')

# Average Speed
fig = plt.figure(figsize=(4, 2.5))
ax = fig.add_subplot(1, 1, 1)
ax.set_title('Average Speed')

m1 = df1.groupby(['Hour'])[['Trip_Seconds']].mean() / 30
m1 = m1.rolling(12).mean().dropna()
lower, upper = m1 - 1.96 * m1.rolling(12).std(), m1 + 1.96 * m1.rolling(12).std()
plt.plot(lower, color='blue', linewidth=1.5, label='2019')
upper = m1 + 1.96 * m1.rolling(12).std()
lower = m1 - 1.96 * m1.rolling(12).std()
plt.plot(upper, color='blue', linewidth=1.5)
lower, upper = m2 - 1.96 * m2.rolling(12).std(), m2 + 1.96 * m2.rolling(12).std()
plt.plot(lower, color='red', linewidth=1.5, label='2022')
upper = m2 + 1.96 * m2.rolling(12).std()
plt.plot(upper, color='red', linewidth=1.5)

ax.set_xlabel('Time (Hour)')
ax.set_ylabel('Average Speed (mph)')
```

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

Thanks for your attention!

Any Questions?

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

About me:

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



南方科技大学

SUSTech

Southern University
of Science and
Technology