Final Project: Analysis and Visualization of Beijing Taxi Trajectories

for the course of

Data Visualization

by
Alejandro Alonso Sánchez

NUA: `147668`
correo: `a.alonsosanchez@ugto.mx`
entrega: `05/06/2025`

Instructor:

Dr. Salvador Botello Aceves

División de Ingenierías Campus Irapuato Salamanca
Universidad de Guanajuato

Contents

Abstract

This research presents an in-depth analysis of taxi trajectory data from Beijing, focusing on developing an interactive visualization system. Working with a dataset containing over 17 million GPS points from 10,357 taxis, I developed a time-based visualization tool incorporating dynamic filtering capabilities. The implementation addressed significant challenges in data processing, spatiotemporal analysis, and interactive visualization optimization. This report details the methodological approach, technical challenges encountered, and solutions developed throughout the project lifecycle.

1 Introduction

The analysis of urban mobility patterns through taxi trajectories represents a crucial aspect of modern urban planning and traffic management [1]. In this project, I focused on creating an interactive visualization system that allows for detailed examination of Beijing taxi movements across different temporal windows. The primary objective was to develop a robust system capable of handling large-scale spatiotemporal data while providing meaningful insights into urban mobility patterns.

2 Dataset Overview

The dataset comprised:

- 10,357 individual taxi trajectory files
- 17,662,984 raw GPS coordinates
- Temporal resolution: Average 30-second intervals
- Spatial coverage: Beijing metropolitan area
- Data fields: taxi_id, timestamp, longitude, latitude

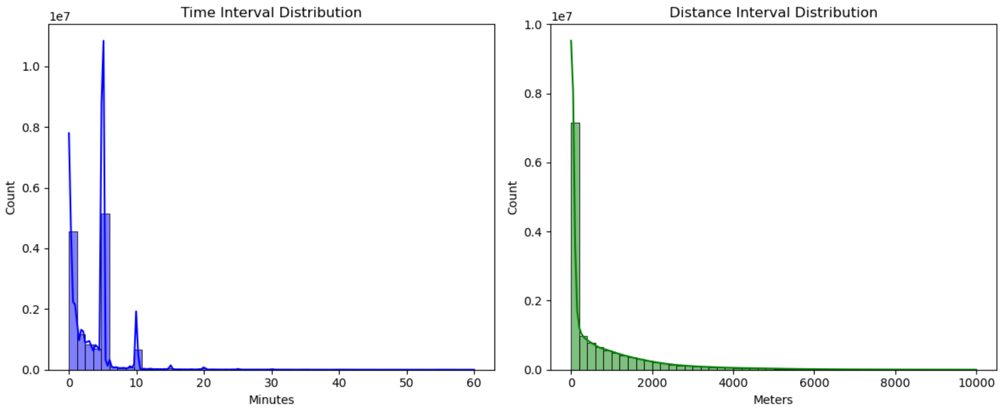


Figure 1: Distributions of distance and time intervals (raw data)

3 Technical Implementation

3.1 Data Preprocessing Pipeline

I implemented a comprehensive preprocessing pipeline:

Algorithm 1 Data Preprocessing Algorithm

```
Require: Raw GPS coordinates dataset $\mathcal{D}$
Ensure: Cleaned and cached dataset $\mathcal{D}'$
Load raw GPS coordinates                                   # Read from file or database
Filter invalid coordinates                                 # Remove NaN/zero values
Convert timestamps to datetime objects                     # Ensure proper time handling
Calculate inter-point distances using Haversine formula    # Great-circle distance
Remove outliers where $\Delta t > 60$ min or $\Delta d > 10$ km   # Threshold-based filtering
Cache processed results $\mathcal{D}'$                     # Save for future use
```

3.2 Key Implementation Challenges

During development, I encountered and resolved several significant challenges:

3.2.1 Data Volume Management:

Processing 17M+ points initially took 3+ hours. I implemented a parallel processing solution using multiprocessing, reducing processing time to 45 minutes. Caching the data in a pickle format reduces reading the data to around 20s.

3.2.2 Visualization Performance:

Direct plotting of millions of points caused significant lag. I implemented Datashader [3] for efficient rendering, achieving more fluid interactivity.

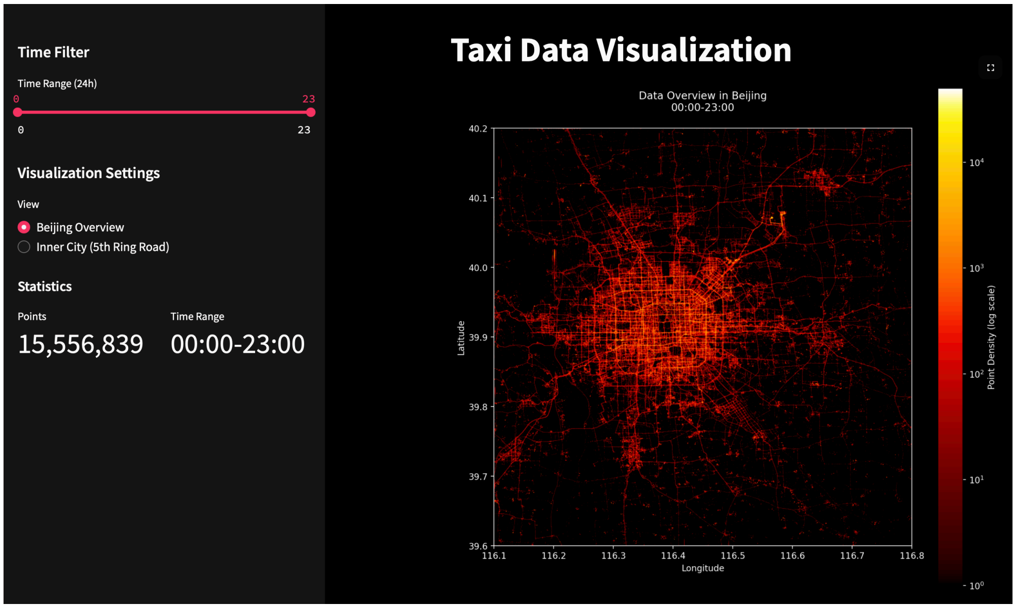
4 Results

4.1 Visualizer

The visualizer reads the data once and caches the Taxi density across Beijing, then it offers a smooth user experience for changing the parameters. The streamlit [2] library allows for flexibility of implementations of the interface.

4.1.1 Time windows

The slider has two ends that can be adjusted for any time period with an hour-to-hour precision.



Time Filter

Time Range (24h): 0 – 23

Visualization Settings

View

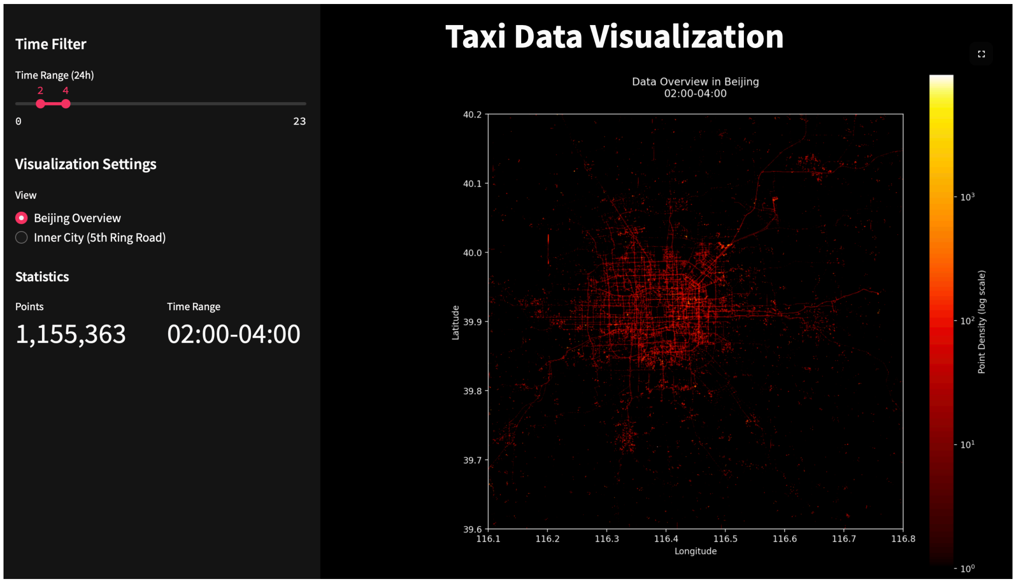
● Beijing Overview
○ Inner City (5th Ring Road)

Statistics

Points: 15,556,839

Time Range: 00:00-23:00

Figure 2: Time windows (full)



Time Filter

Time Range (24h)

2 4

0 23

Visualization Settings

View

● Beijing Overview

○ Inner City (5th Ring Road)

Statistics

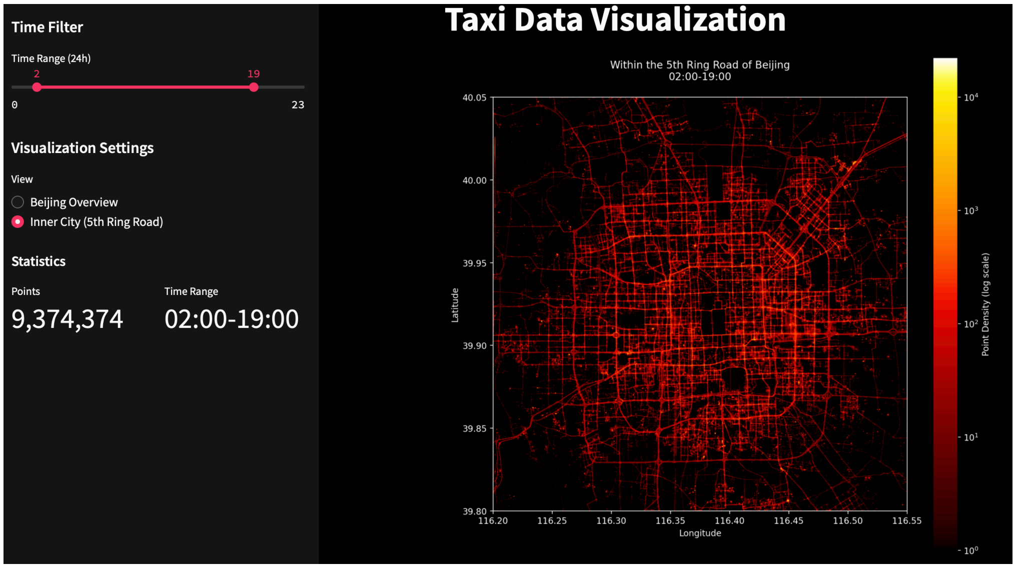
Points	Time Range
1,155,363	02:00-04:00

Taxi Data Visualization

Figure 3: Time windows (reduced)

Notice the statistics and number of points are reduced, as well as the intensity of the heatmap.

4.1.2 Two view modes: city overview and inner city (5th Ring Road)



Time Filter

Time Range (24h)

2 19

0 23

Visualization Settings

View

○ Beijing Overview

● Inner City (5th Ring Road)

Statistics

Points	Time Range
9,374,374	02:00-19:00

Taxi Data Visualization

Figure 4: Inner city view

4.2 Statistical Analysis

4.2.1 Cleaned distributions

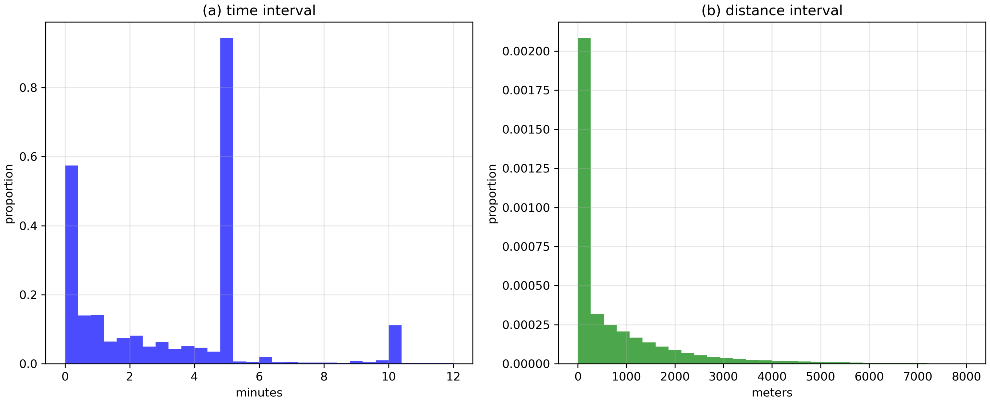


Figure 5: Interval distributions (cleaned)

4.2.2 Points per sample

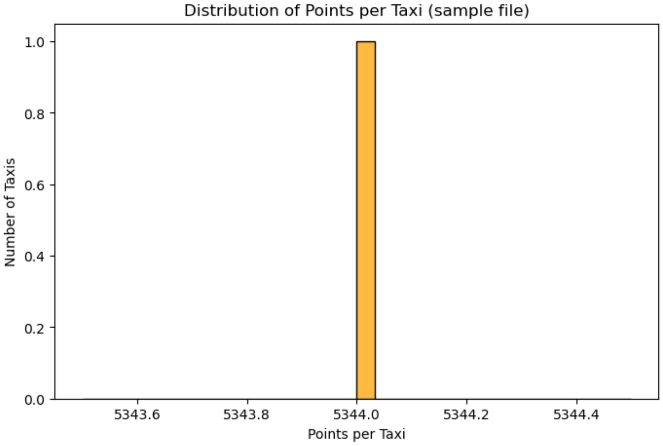


Figure 6: Points per taxi

4.2.3 Speed overview

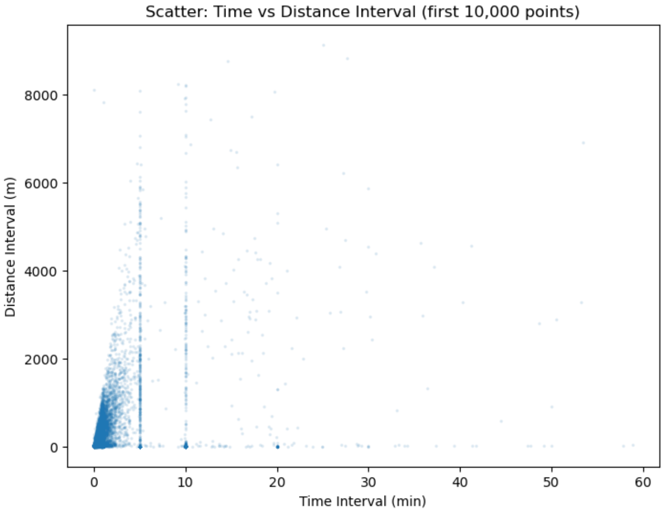


Figure 7: Scatter: time vs distance

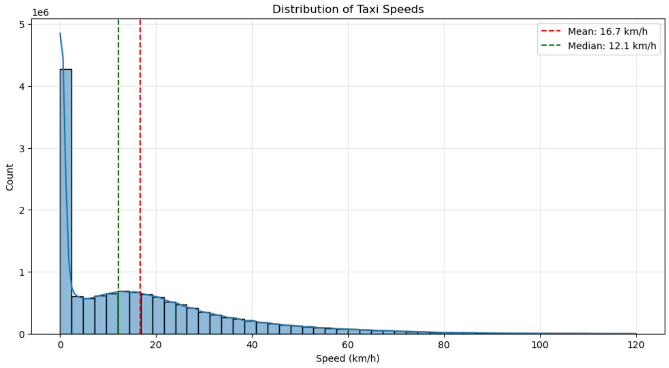


Figure 8: Speed

Key findings from the data:

- Average time interval between points: 3.6 minutes
- Average distance between points: 750 meters
- Peak activity hours: 2pm - 10pm
- Average speed 16.7 km/h which is in line for big cities with heavy traffic
- Standard deviation of 18.3 km/h represents high variability (mix of standstills and brief high-speed segments).
- Highest density areas: Central Business District

5 Future Work

Potential improvements include:

- Implementation of trajectory prediction
- Integration with traffic data

- Analysis of seasonal patterns
- Machine learning-based pattern recognition
- Implementation of taxis trajectory and common trips with trip id dataset

References

[1] Castro, P. S., Zhang, D., Chen, C., Li, S., & Pan, G. (2013). From taxi GPS traces to social and community dynamics: A survey. ACM Computing Surveys.

[2] Streamlit Team (2023). Streamlit Documentation: Building Data Apps.

[3] Datashader Development Team (2023). High-Performance Python Visualization Toolkit.

[4] Wang, X., et al. (2020). Understanding urban mobility patterns through taxi trajectory mining.