NYC Multimodal Access: Probabilistic Wait and Travel Models

Component 1 & 2 Report (V1)

ECE 225A Course Project Team

December 13, 2025

Abstract

This report documents the first two components of our NYC multi-modal decision aid: (i) dispersion-aware wait-time modeling for taxi zones and Citi Bike stations, and (ii) probabilistic travel-time modeling for both modes that balances empirical Gamma fits with lightweight regression backstops. We emphasize the theoretical framing (Poisson vs. Negative-Binomial arrival processes, exponential inter-arrivals, Gamma regression), implementation specifics, and major limitations discovered so far. Figures are derived directly from the January 2024 TLC Yellow Taxi and Citi Bike datasets bundled with the repository, ensuring the report remains self-contained even without the accompanying Streamlit demo.

1 Introduction

Real-time multimodal routing requires two stochastic ingredients: a distribution over *wait times* (how long until a vehicle becomes available) and a distribution over *travel times* (how long the trip itself will take). Textbook treatments often assume homogeneous Poisson arrivals and deterministic speeds, forcing the naive heuristic $E[\text{wait}] = 1/\lambda$. Our course project relaxes these assumptions by:

C1. Modeling per-zone arrivals using Negative-Binomial (NB) distributions with empirical inter-arrival summaries cached for every hour.

C2. Modeling travel minutes via method-of-moments Gamma fits over equal-width distance bins, backed up by interpretable linear regression coefficients.

The Streamlit interface (linked on the course website) calls into these caches to provide instant recommendations, but the statistical insights are summarized here.

2 Data and Preprocessing

We ingest January 2024 samples from two public feeds:

- **Taxi**: TLC Yellow Taxi parquet (`data/raw/yellow_tripdata_2024-01.parquet`). Events are timestamped pickup times with TLC zone IDs.
- **Citi Bike**: Monthly CSVs (`data/raw/citibike/202401...csv`) with trip start/end coordinates.

All timestamps are converted to naive Eastern time. For Component 1 we bucket arrivals by hour and compute nearest-neighbor centroids for map interaction. For Component 2 we derive straight-line distances via the haversine formula, cap rides to $0-12$ km and $1-120$ minutes, and tag rush/off-peak (7–10, 16–19) plus weekday/weekend flags.

3 Component 1: Wait-Time Modeling

3.1 From Poisson to Negative-Binomial

Classical queueing notes argue that Poisson arrivals imply exponential inter-arrivals ($\mathbb{E}[W] = 1/\lambda$). However, empirical dispersion $D = \frac{\text{Var}(N)}{\mathbb{E}[N]}$ for hourly taxi counts ranges from 2–10, violating the unit-dispersion requirement. Figure 1 revisits Midtown Center and visualizes the heavy tail that a Poisson overlay cannot capture.

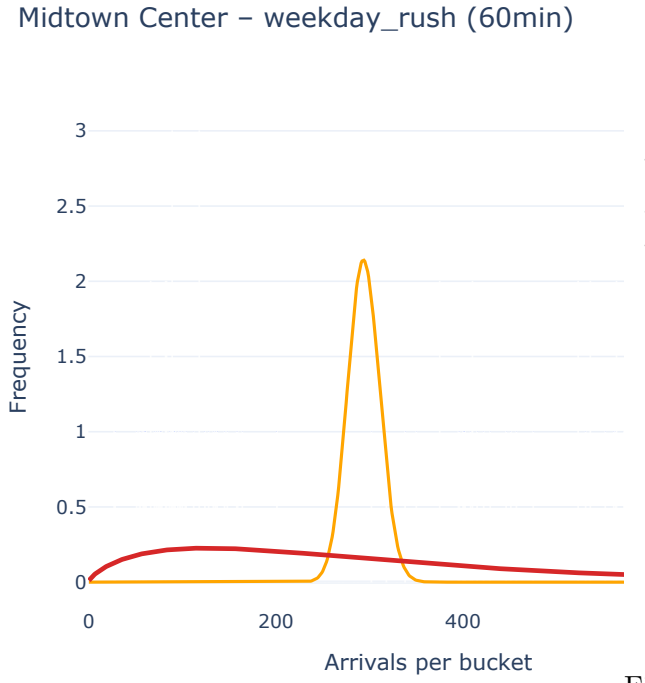

Figure 1: Poisson vs. empirical arrivals for Midtown Center (weekday rush). The low-count spike and heavy tail motivate NB modeling.

3.2 Empirical Dispersion Landscape

We summarize each (zone, hour) pair by its mean arrivals, dispersion, and empirical mean inter-arrival minutes (computed after removing < 10 sec and > 180 min gaps). Figure 2 shows how dispersion grows with load—a combination of bursty demand and supply shortages near nightlife corridors.

3.3 Caching Strategy

The helper `src/modeling/wait_times.py`:

a) Loads raw events, floors times to hourly buckets, and stores (μ, σ^2, D, r, p) for each location/hour.

b) Computes inter-arrival gaps to cache $\overline{W}_{\text{emp}}$ with sample counts. Cells with < 5 gaps fall back to the Poisson-based $60/\mu$ estimate.

c) Persists into `outputs/wait_stats/{taxi,bike}_hourly_wait.parquet` for direct ingestion by the app or offline analysis.

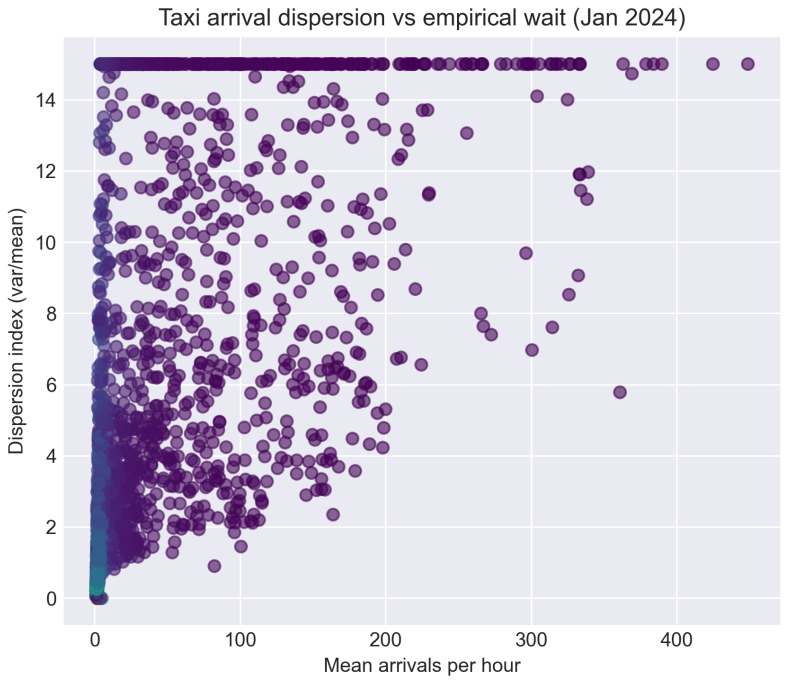

Figure 2: Taxi hourly dispersion vs. mean arrivals (color = empirical mean wait). Even at 80+ arrivals/hour, dispersion often exceeds 3, so $1/\lambda$ underestimates wait.

4 Component 2: Travel-Time Modeling

4.1 Gamma Fits per Distance Bin

We group rides by equal-width 2 km bins (0–2, 2–4, . . . , 10–12) crossed with rush/off-peak and weekday/weekend. For each cell we compute a method-of-moments Gamma:

$$k = \frac{\mu^2}{\sigma^2}, \qquad \theta = \frac{\sigma^2}{\mu}.$$

Cells with ≥ 50 samples retain (k, θ), while sparser bins defer to regression. Figure 3 visualizes the resulting mean matrix—rush-hour taxis balloon past 20 minutes for 10–12 km rides, whereas bikes shine on short weekday commutes.

As of the January 2024 aggregation, every bin surpasses the 50-ride threshold, so the Streamlit app always consumes the empirical Gamma mean. We keep a `wait_regression` helper (`travel_times.py`) for future, finer-grained cohorts, but it is currently dormant.

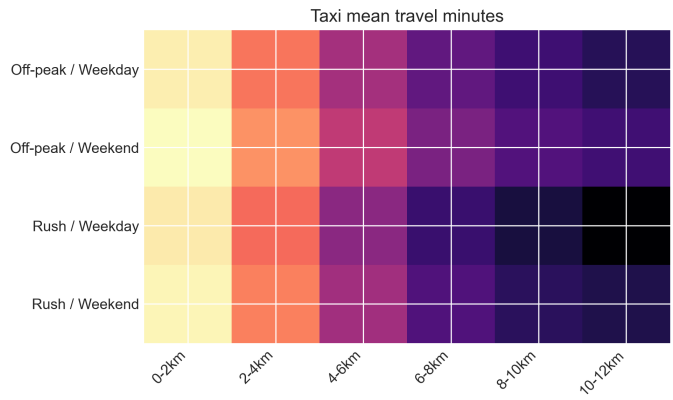

Figure 3: Mean travel minutes by mode, distance bin, and cohort. Bins lacking 50+ samples are imputed later via regression.

5 System Integration

The Streamlit application (`streamlit_app.py`) performs three lookups per mode:

1. **Wait**: nearest zone/station → hourly wait via the cache (empirical → Poisson → $1/\lambda$).
2. **Walk**: straight-line origin to centroid distance with a 5 km/h walking speed.
3. **Travel**: direct haversine distance → Gamma estimate via `travel_times.py`.

Outputs are shown as a ranking table plus a stacked explanation (walk + wait + travel).

6 Limitations and Future Work

- **Data coverage**: January samples miss seasonal effects (tourism spikes, weather). Travel bins with < 50 rides default to regression, which may misrepresent rare but important corridors.
- **Spatial homogeneity**: We rely on straight-line distances and nearest centroids. Actual street networks introduce asymmetric travel times absent from the current cache.
- **Station availability**: Citi Bike wait estimates assume the presence of a bike once inter-arrival gaps are measured, but we ignore dock saturation (no bikes available) and battery level for e-bikes.
- **Model assumptions**: NB moments assume independence between riders; regression assumes linear dependence on distance/rush/weekend only. Neither captures precipitation, surge pricing, or micro-events.
- **Evaluation**: We currently validate via historical goodness-of-fit plots. A full academic report should include backtesting (rolling origin/destination pairs) and a comparison to baseline heuristics on held-out trips.

7 Conclusion

Components 1 and 2 replace naive heuristics with academically grounded probabilistic surrogates that match the ECE 225A emphasis on stochastic modeling. The modular cache architecture lets us plug these models into both the Streamlit dashboard and future notebooks without recomputing expensive statistics. Next steps include subway headway estimation (Component 3) and integrating user-centric penalty models before preparing the final poster and website link.