Network Modeling Across NYC Traffic Patterns

Visual Methods for Interpreting Complex Graphs

Sean Fitch
Data Analytics CSCI-4600

Problem Statement

- Urban transportation networks are complex and challenging to visualize
- Core Challenge: Create a workflow for transforming raw network data into an interpretable graph
    - Can help to quickly find patterns in data
    - Can help convey results to audiences
    - Ideally would generalize to many network types

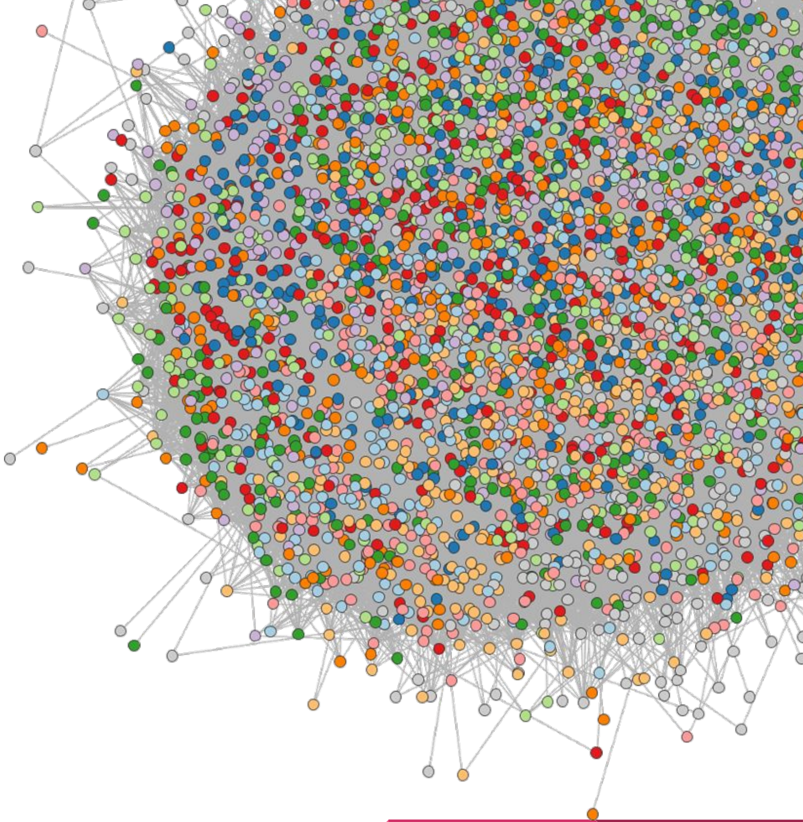


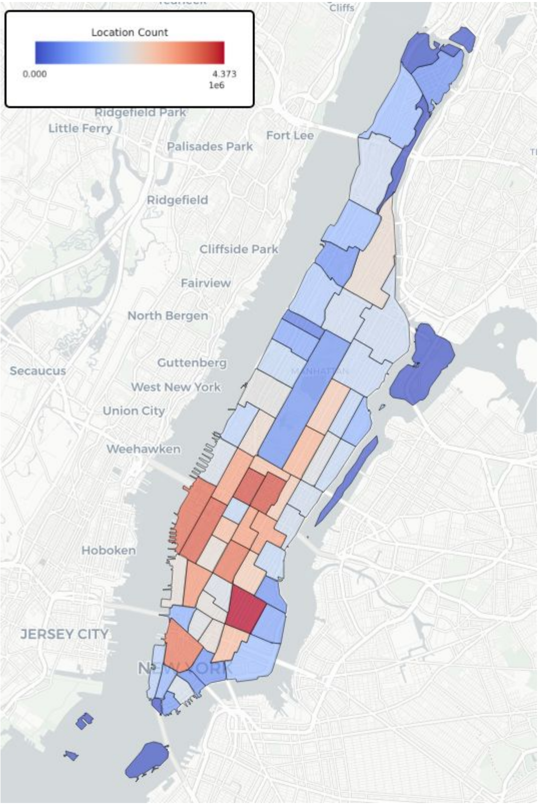
Image: Brian Staats, https://observablehq.com/@bstaats/graph-visualization-introduction

Dataset Overview

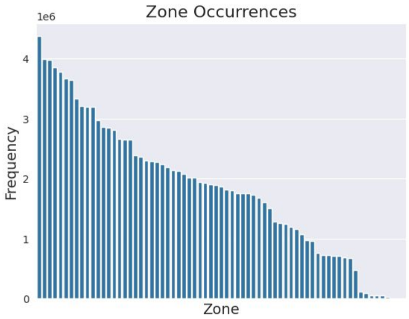
- New York City taxi zone network:
  - Billions of trips recorded by Taxi and Limousine Commission
  - Intricate interconnections resist simple representation
  - Original Kaggle dataset: 60+ GB as binary (2011-2023)
- Data Processing:
  - Filtered to Manhattan trips in 2022 from High-Volume For-Hire Vehicle (FHV) services
  - Drop Extraneous columns
  - Final dataset: 63,110,170 trips
- Key Columns:
  - Pickup and Dropoff Location IDs
  - Trip miles, time, and fees
  - Service providers (Uber, Lyft, Via, Juno)

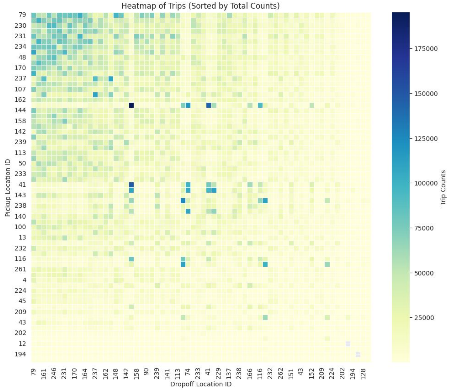
Image: NurPhoto / Getty Images

Preliminary Analysis: Spatial Distribution



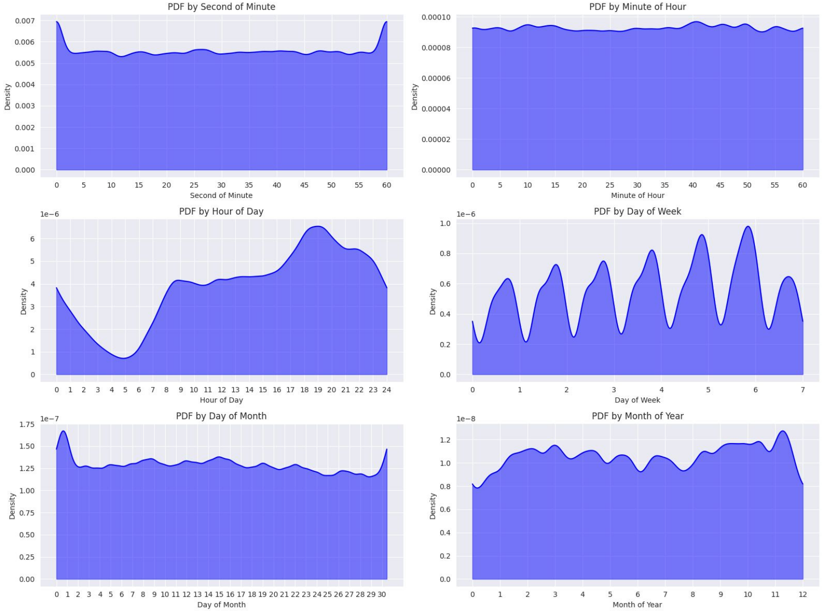
- Significant variation in trip frequencies
- Ranges from 0 trips (Governor's Island) to 4.37 million trips (East Village)
- Occurrences has linear, pickup-dropoff pairs have quadratic distribution





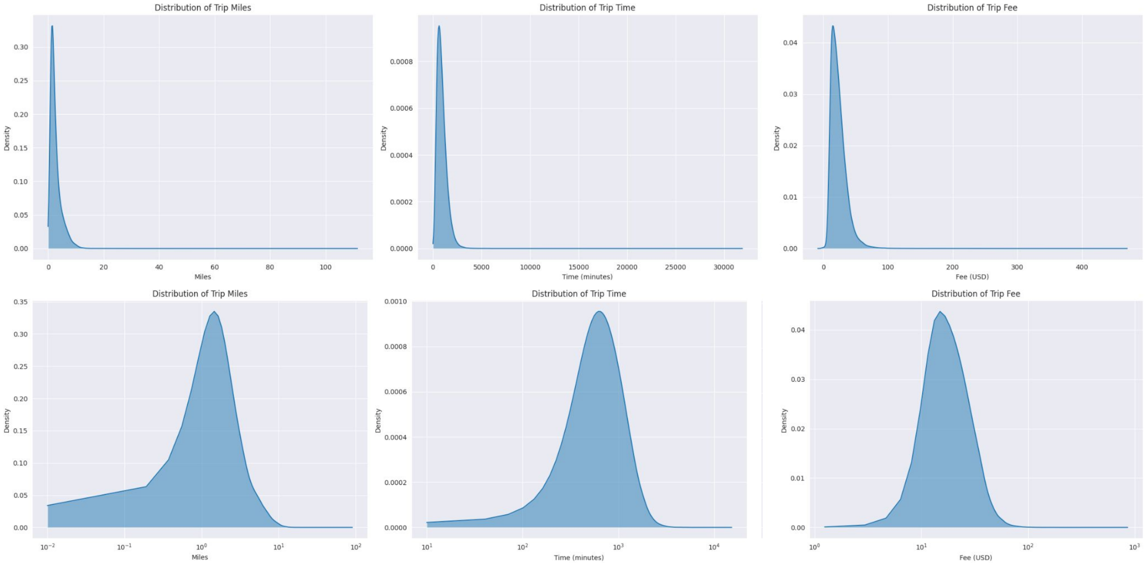
Preliminary Analysis: Temporal Patterns

- Minute/Hour: Significant non-uniformity
  - Minute chi-square: 440k, p-val 0.0
  - Excluding 0 second: 1k, p-val 1e-189
  - Hour: 52k, p-val 0.0
  - 5 minute: 32k, p-val 0.0
- Daily: Dip between 1-7 AM. Peak at 7 PM.
- Weekly: Increasing throughout the week.Drops on Saturday
- Monthly: More trips on the 1st. About weekly peaks.
- Yearly: Irregular. Lowest in January, highest in early December.



Preliminary Analysis: Other Variables

- Fee, time, and miles have similar distributions
  - Close to log-normal
  - Likely have strong correlation
- For some reason, base fare and driver pay can be negative.



Distribution of Trip Miles

Distribution of Trip Time

Distribution of Trip Fee

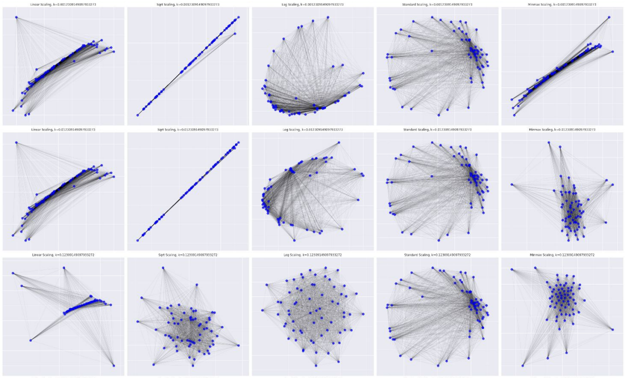
Distribution of Trip Miles

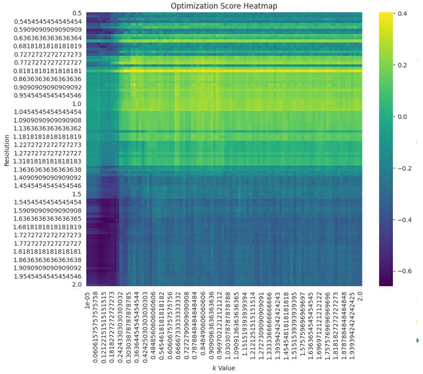
Distribution of Trip Time

Distribution of Trip Fee

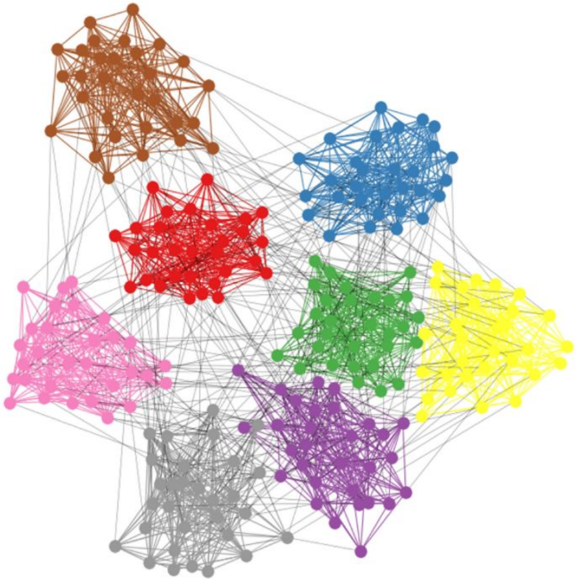
Graph Visualization: Early Attempts

- Grid search for layout, scaling, and parameters
  - Spring, Kamada Kawai, shell, Fruchterman-Reingold
  - Linear, Log, Sqrt, Standard, Minmax
  - Thresholding & k
  - No Results produced visually separable clusters
- Clustering algorithms for coloring
  - Louvain, Girvan-Newman, Infomap, Label Propagation, Spectral
  - Doesn't improve the underlying problem of layout
- Grid search on clustering and layout parameters and grade results on silhouette of clusters
  - Produced imbalanced or few clusters
  - Increasing number of parameters became difficult to tune
  - Scores depend far more on initial clustering than on the actual layout of the graph





Layout Optimization



- A good network visualization is one which:
  - Has strong *visually* separable features
  - These features respect the underlying graph characteristics
- Measure visual features in network visualizations using spatial clustering algorithms
  - Identify spatially distinct groups of nodes (clusters).
  - Use techniques like K-means or DBSCAN.
- Measure graph characteristics with clusters based on edge weights.
  - Assess if strongly connected nodes are placed close together.
- Evaluate graphs based on alignment of the two cluster sets

Image: Dang, T.D., Do, D.H. & Phan, T.H.D. https://doi.org/10.1007/s13278-023-01080-1

Toolset for Layout Optimization

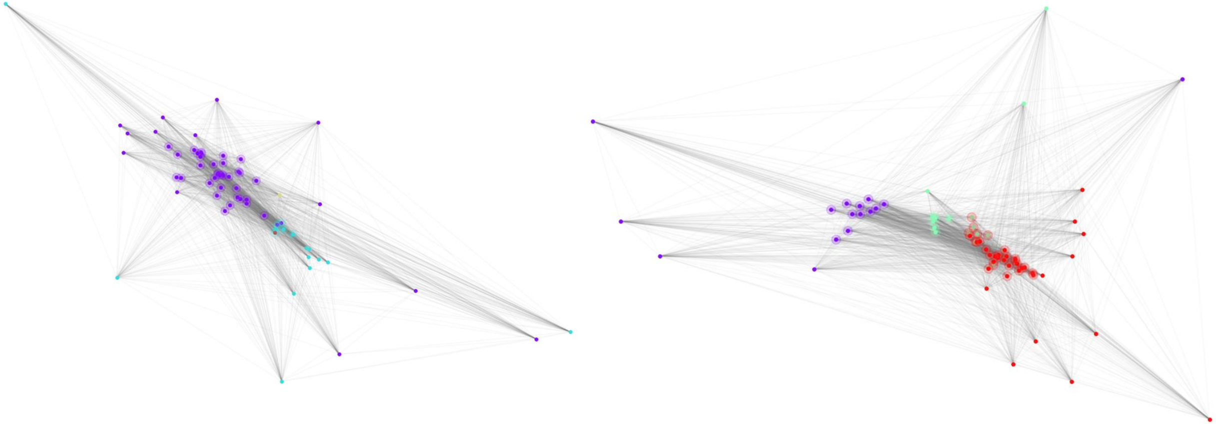
- Louvain Clustering
    - Graph-based community detection algorithm that aims to partition nodes in a graph into clusters s.t. edges within communities are dense, and edges between communities are sparse.
- HDBSCAN (Hierarchical Density-Based Spatial Clustering of Applications with Noise)
    - Clustering algorithm designed to group data points into clusters based on density, robust to arbitrary shapes and densities.
- Fruchterman-Reingold algorithm
    - Graph layout algorithm used to position nodes in a graph in a visually appealing manner, by simulating nodes as charged particles and edges as springs.
- Fowlkes-Mallows Index (FMI)
    - Metric used to assess the similarity between two clusterings. It measures the quality of clustering results by comparing the pairs of data points assigned to the same cluster in both the true clustering and the clustering to be evaluated. Range 0-1. (Label agnostic)

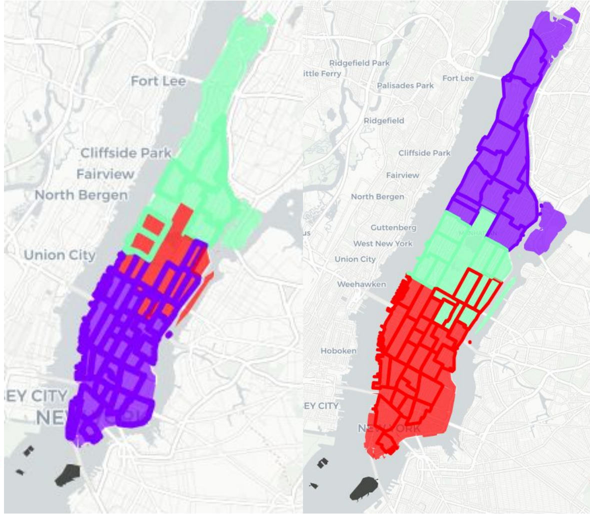
Optimization Approach

Input: Weight matrix, number of clusters (n)

1) Grid search over resolution to find a Louvain Clustering which produces n clusters
2) Grid search over k and apply HDBSCAN
    a) Calculate FMI score between two cluster sets
    b) Store the max FMI layout
3) Return max FMI and clusters for each method
4) Find pairing of clusters from each algorithm which maximizes overlap
    a) Integer programming to maximize trace of confusion matrix by column permutations
5) Plot the network, color nodes by each cluster.

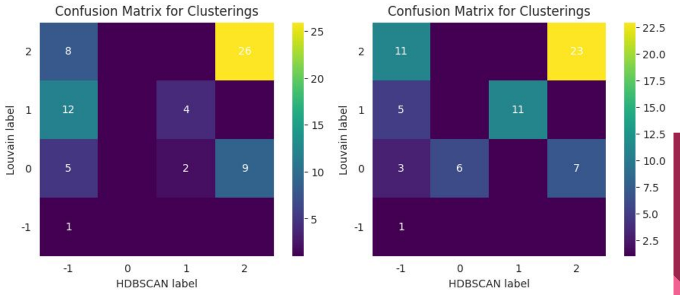
Results: Cluster Visualization





Left/Right: Results with/without optimization on the same set of Louvain clusters

Clusters are visually shown in optimized network



Future Work

- Improving optimization:
    - Apply methods to determine best number of clusters / Louvain parameters
    - Apply grid search to HDBSCAN parameters
    - Optimize all grid searches using heuristics and other search algorithms
    - Try going back to using silhouette or other metrics on only Louvain clusters
        - Prevent added complexity of HDBSCAN step
        - May unnecessarily skew results away from layouts that have irregularly shaped features
- Testing optimization:
    - More complex networks (All NYC taxi data?)
    - Disparate networks
- Original research question:
    - Where should NYC focus construction efforts?