

NYC 311 Insights Challenge — Insight Report

1. Executive Summary

This project addresses the NYC 311 Insights Challenge, where the goal is to forecast daily 311 call volume, detect anomalies (spikes and dips), and provide an interactive dashboard for city officials. By anticipating service demand, the solution helps improve staff allocation, reduce overtime costs, and prepare for sudden surges such as holidays, weather events, or public gatherings.

2. Data & Windows

The dataset is sourced from NYC Open Data's 311 Service Requests API. Requests were aggregated into daily totals per borough. The challenge specifies two windows: • Training & feature engineering: August 1, 2024 – April 30, 2025 • Forecasting/scoring horizon: May 1, 2025 – August 1, 2025. Optional external features (e.g., holidays, weather) must be dated before April 30, 2025.

3. Methodology

The pipeline includes the following steps:

- Data ingestion: Daily call totals computed using SoQL API queries.
- Feature engineering: Calendar-based features (day-of-week, month, cyclic encodings), trend features, and borough splits.
- Forecast model: Gradient boosting (XGBoost) blended with weekday-average baseline for stability.
- Anomaly detection: Percent deviation = $(\text{Actual} - \text{Expected}) / \text{Expected}$; top-5 spikes and dips flagged.

4. Results

Metric	Value
Validation RMSE (last 28d)	≈ 553
Validation MAPE	≈ 4.9%
Top anomalies	Holidays, storms, and sudden dips during weekends

5. Dashboard

A Streamlit dashboard was developed with the following features: • NYC borough map (token-free, Plotly mapbox). • Time-series Actual vs Forecast with anomalies. • Borough mix stacked area chart. • Day-of-Week heatmap. • KPIs and filters (date range, borough, moving average). This tool enables non-technical users to explore patterns interactively.

6. Reproducibility

The repository ensures strict reproducibility with: • One notebook (`submission.ipynb`) that runs end-to-end on CPU in under 10 minutes. • Outputs `submission.csv` and `anomalies.csv` with exact column specs. • Requirements pinned in `requirements.txt`. • README with two copy-paste blocks: environment setup and run instructions. • `Data/train.csv` and `data/test.csv` excluded via `.gitignore` (organizers provide them).

7. Conclusion

This project demonstrates a complete, reproducible pipeline for forecasting NYC 311 service call volume, identifying anomalies, and providing a decision-support dashboard. The approach balances accuracy, interpretability, and real-time usability, directly supporting operational efficiency for the city.