Data Bootcamp Midterm Project

How Safe is NYC?

Exploratory Data Analysis with the NYPD Crime API

Will Wu - John Yue

October 23, 2025

Abstract

This project explores patterns in New York City crime reports using the *NYPD Complaint Data (Historic)* from NYC Open Data. We focused on writing clear, reproducible Python code and interpreting real-world data through visualization and analysis. Starting from the Socrata API, we cleaned and explored over a million records, created new features for time and severity, and visualized complaint patterns across boroughs. We also built a simple metric called the *Relative Safety Index (RSI)* to summarize how "safe" each borough appears based on complaint volume and average severity. Our goal is not to make causal claims but to practice end-to-end analysis, from API access to final insights, in a way another student could fully reproduce.

1 Introduction

For our midterm project, we wanted to apply everything we learned in class to a real dataset that felt relevant to where we live. Our question was simple but open-ended: *Which parts of New York City appear safer based on reported police complaints?*

We chose this topic because safety is something everyone in NYC thinks about, yet the data behind it are often misunderstood. Our goal was to write a clean and reproducible notebook that any student could follow to explore similar questions. Instead of aiming for a professional or academic tone, we focused on transparency, reproducibility, and readable code that shows our reasoning at every step.

2 Dataset and API Access

We used NYC Open Data's *NYPD Complaint Data (Historic)* (dataset ID `qgea-i56i`), which is publicly available through the Socrata API. Using the API helped us avoid manual downloads and made our workflow reproducible.

To make the analysis more manageable, we selected only a few key columns:

- `cmplnt_fr_dt`: date of the incident
- `ofns_desc`: offense description
- `law_cat_cd`: offense severity (felony, misdemeanor, violation)
- `boro_nm`: borough name

- `addr_pct_cd`: precinct
- `lat_lon`: location coordinates

This subset still includes a mix of numerical, categorical, and geospatial data, which satisfies the project requirements and keeps runtime reasonable.

3 Setup and Cleaning

We started by installing essential Python packages: `pandas`, `matplotlib`, `seaborn`, `sodapy`, and `geopandas`. Consistent Seaborn styles and figure sizes ensured that all our graphs were clean and readable.

Once the data was loaded, we cleaned and formatted it for analysis:

- Converted `cmplnt_fr_dt` to proper `datetime`.
- Standardized text fields like borough names to prevent duplicates (e.g., "BROOKLYN" vs. "Brooklyn").
- Extracted numeric latitude and longitude from the JSON-like `lat_lon` field.
- Filtered coordinates to within NYC's bounds to remove outliers.

We also engineered a new `severity_score` column, mapping Felony = 3, Misdemeanor = 2, and Violation = 1. This helped us later compute averages and visualize patterns by severity. While not the most exciting step visually, it ensured that our later plots accurately reflected the data.

4 Crime by Borough and Severity

Our first set of graphs focused on overall complaint distribution by borough and offense type.

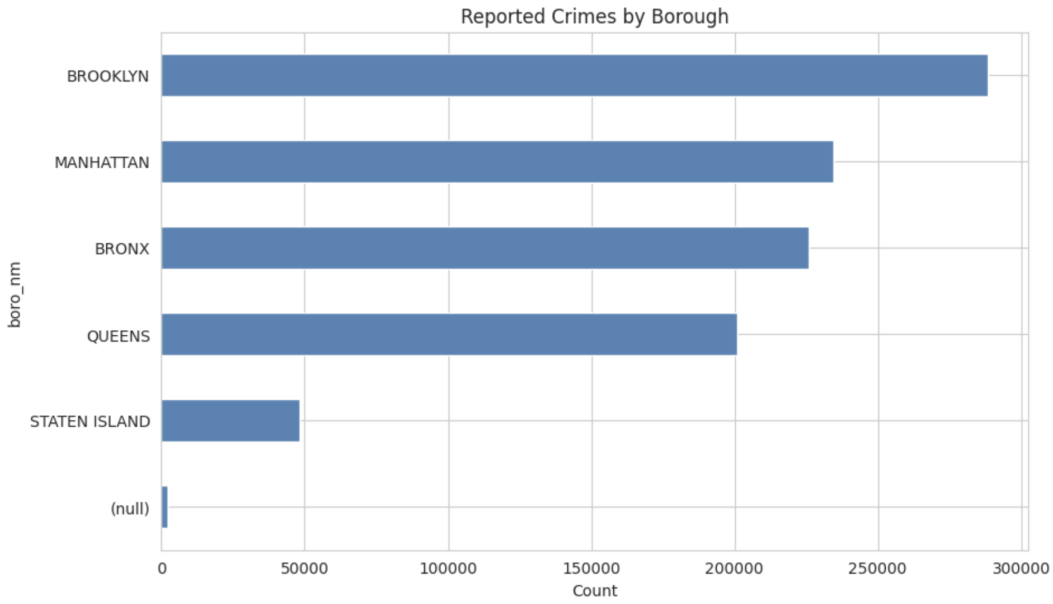

Figure 1: Reported complaints by borough.

Figure 1 shows that Brooklyn and Manhattan lead in complaint volume, followed by the Bronx and Queens, with Staten Island lowest. However, since borough size and activity levels differ greatly, high counts don't automatically mean more danger.

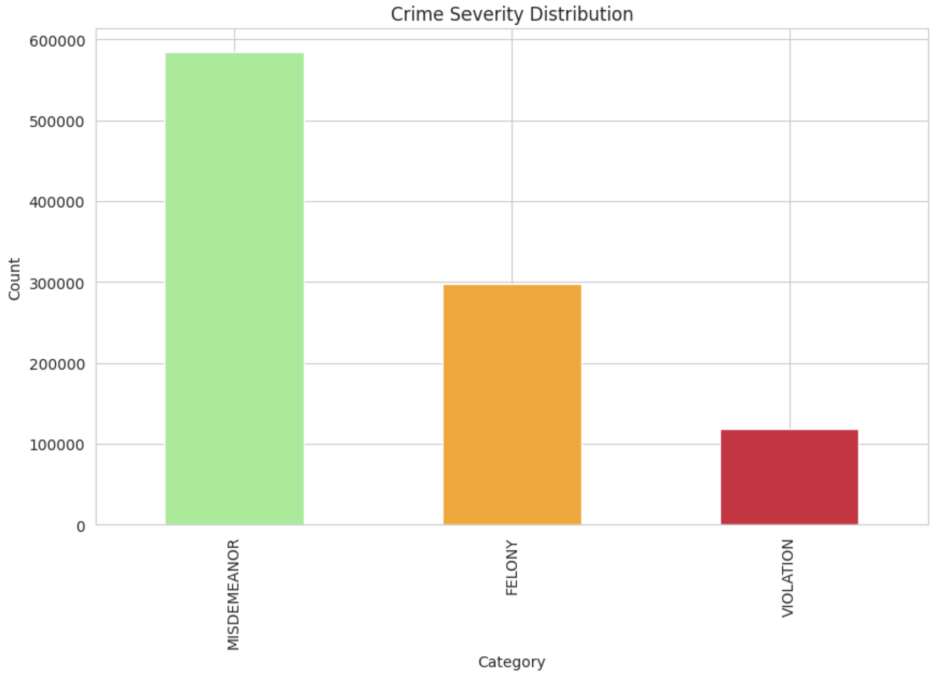

Figure 2: Distribution of complaints by legal category.

As shown in Figure 2, misdemeanors dominate NYC's reports, while felonies and violations make up smaller shares. This suggests most recorded incidents are mid-level offenses rather than violent crimes.

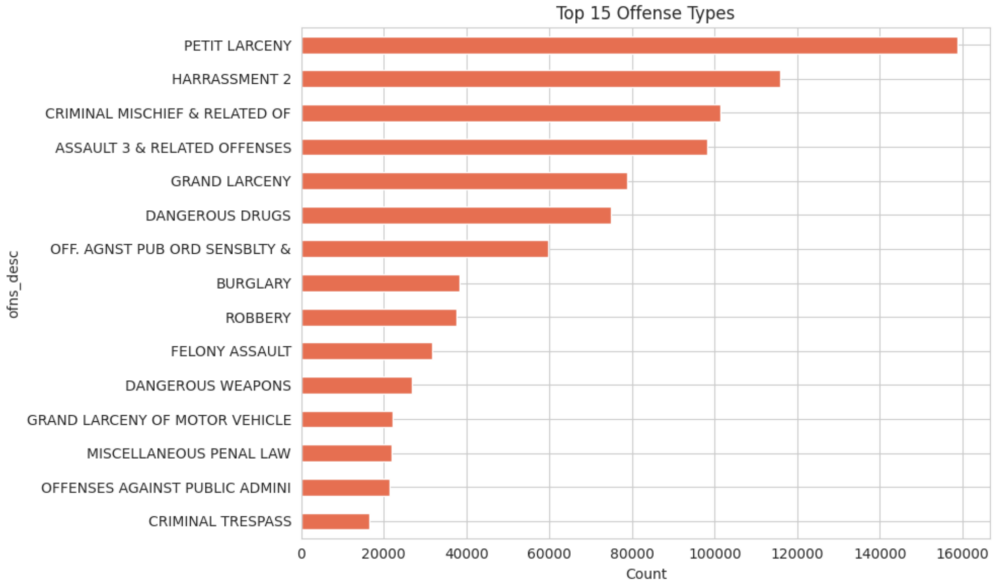

Figure 3: Top 15 offense types by complaint count.

Figure 3 lists the fifteen most common complaint types, led by **Petit Larceny**, harassment, and criminal mischief. Seeing these side by side helped us understand that complaint data often reflects everyday disputes and thefts more than serious violent crimes.

5 Spatial Hotspots with GeoPandas

To visualize where complaints cluster across the city, we used GeoPandas and Contextily to map over one million points. After filtering coordinates within NYC's geographic bounds, we plotted each incident as a red dot.

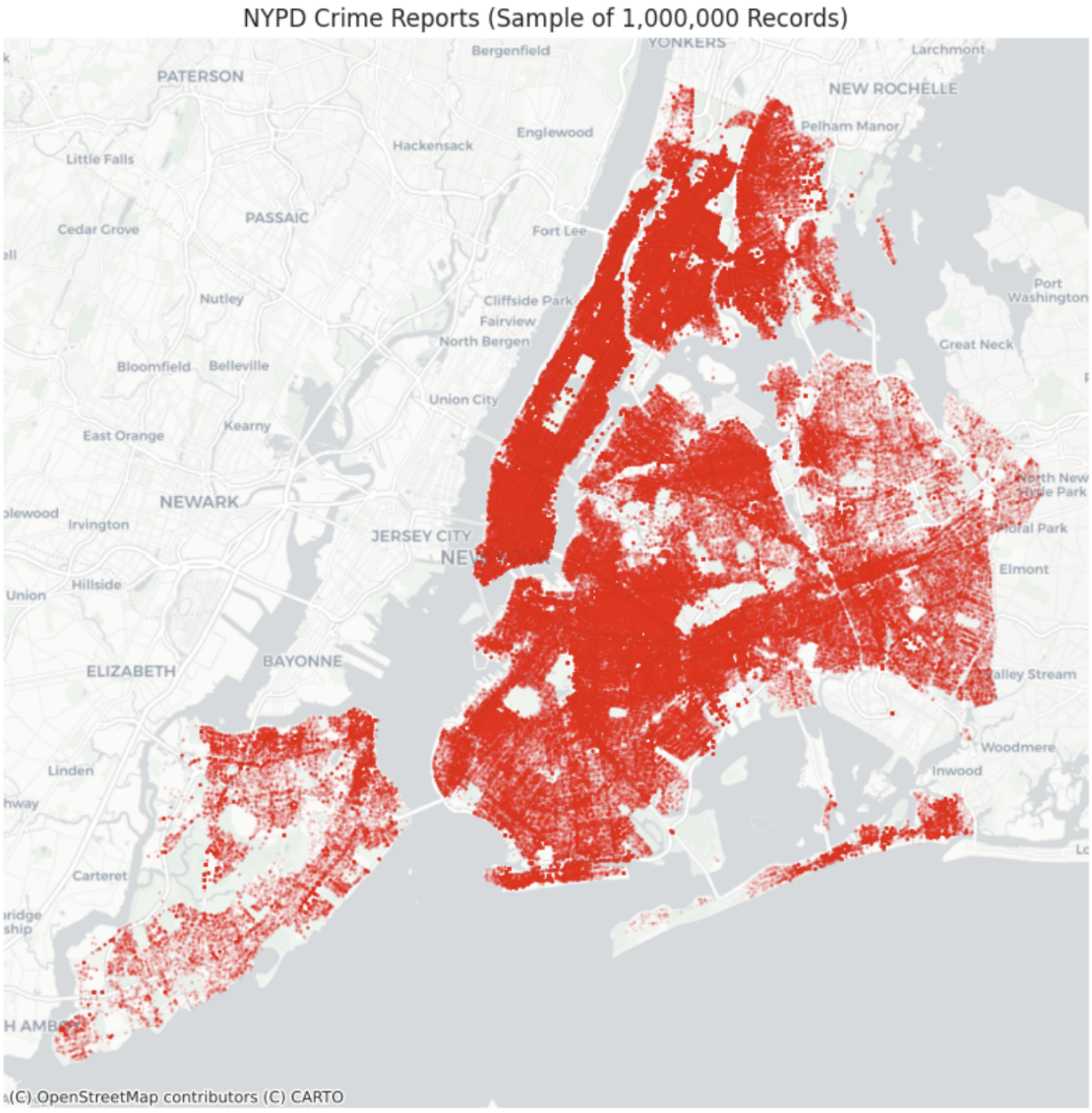

Figure 4: NYPD complaint locations across NYC (sample of 1 million records).

The map (Figure 4) highlights dense clusters around Midtown Manhattan, Downtown Brooklyn, and other high-traffic areas. These align with commercial centers, nightlife zones, and transit corridors—places where large crowds naturally generate more reports. This visualization also re-

minded us of the importance of spatial filtering: removing mis-geocoded points outside NYC made the map readable and relevant.

6 Computing a Relative Safety Index

Next, we wanted a way to compare boroughs on both complaint frequency and severity. We built a composite measure called the *Relative Safety Index (RSI)*:

$$\mathrm{RSI}_b = \frac{1}{\overline{s}_b} \times \frac{1}{(n_b / \max n_b)} \quad \text{(higher = safer)}.$$

This formula rewards boroughs with fewer and less severe complaints.

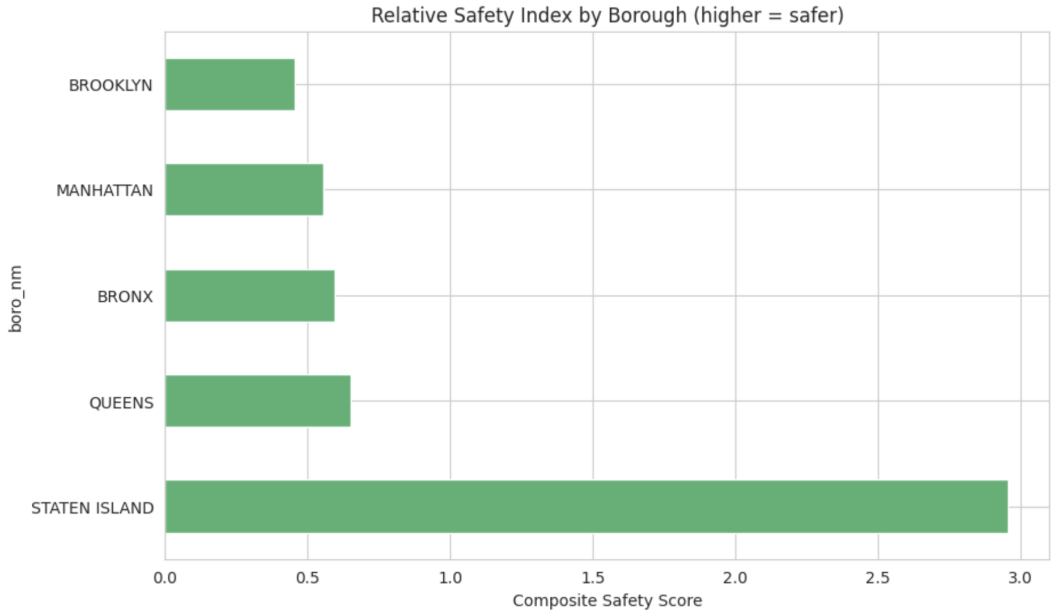

Figure 5: Relative Safety Index (RSI) by borough.

Figure 5 shows the resulting RSI values. Staten Island ranks highest (safest), followed by Queens and the Bronx, while Brooklyn and Manhattan score lower due to their high volume of reports. Although the index is simple, it provides an interpretable metric for comparing boroughs—clear enough that another student could reproduce or modify it easily.

7 Insights from the Visuals

After producing the main figures, several takeaways stood out:

1. Volume doesn't equal risk. Boroughs with more people and tourism, like Manhattan, naturally have more reported incidents. The exposure effect is visible in both the bar chart and map.

2. Mid-level offenses dominate. Most of NYC's reports fall under misdemeanors, meaning overall complaint levels can rise without implying more violent activity.

3. Visualization simplifies complexity. The RSI distills multiple variables—frequency, severity, and geography—into a single readable number. This exercise showed how composite metrics can clarify patterns rather than obscure them.

8 Limitations and Next Steps

Our analysis is descriptive, not causal. Complaint data reflects what is reported, which varies by policing patterns, neighborhood trust, and population density. Without population or foot traffic denominators, raw counts can't be interpreted as absolute risk.

If extended, we would:

- Normalize complaint counts by resident or daytime population.
- Explore time patterns—day vs. night, weekday vs. weekend.
- Add density heatmaps or precinct-level breakdowns.
- Test alternative severity weighting systems or regression-based models.

These extensions would make the index more rigorous and reveal finer-grained safety differences across neighborhoods.

9 Reproducibility and Code Readability

The Colab Notebook attached mirrors this report's structure step by step. Each section (from setup to visualization) is clearly labeled and includes concise comments describing what the code does. Figures use consistent color palettes and labeled axes to make them easy to interpret. Because all data is pulled directly from the public API, anyone can re-run our notebook and obtain identical results, ensuring transparency and reproducibility—two key aspects of good data science practice.

References

NYC Open Data (Socrata). *NYPD Complaint Data (Historic)*. Dataset ID: `qgea-i56i`. https://data.cityofnewyork.us/.
Python libraries used: `pandas`, `numpy`, `matplotlib`, `seaborn`, `sodapy`, `geopandas`, `shapely`, `contextily`.