

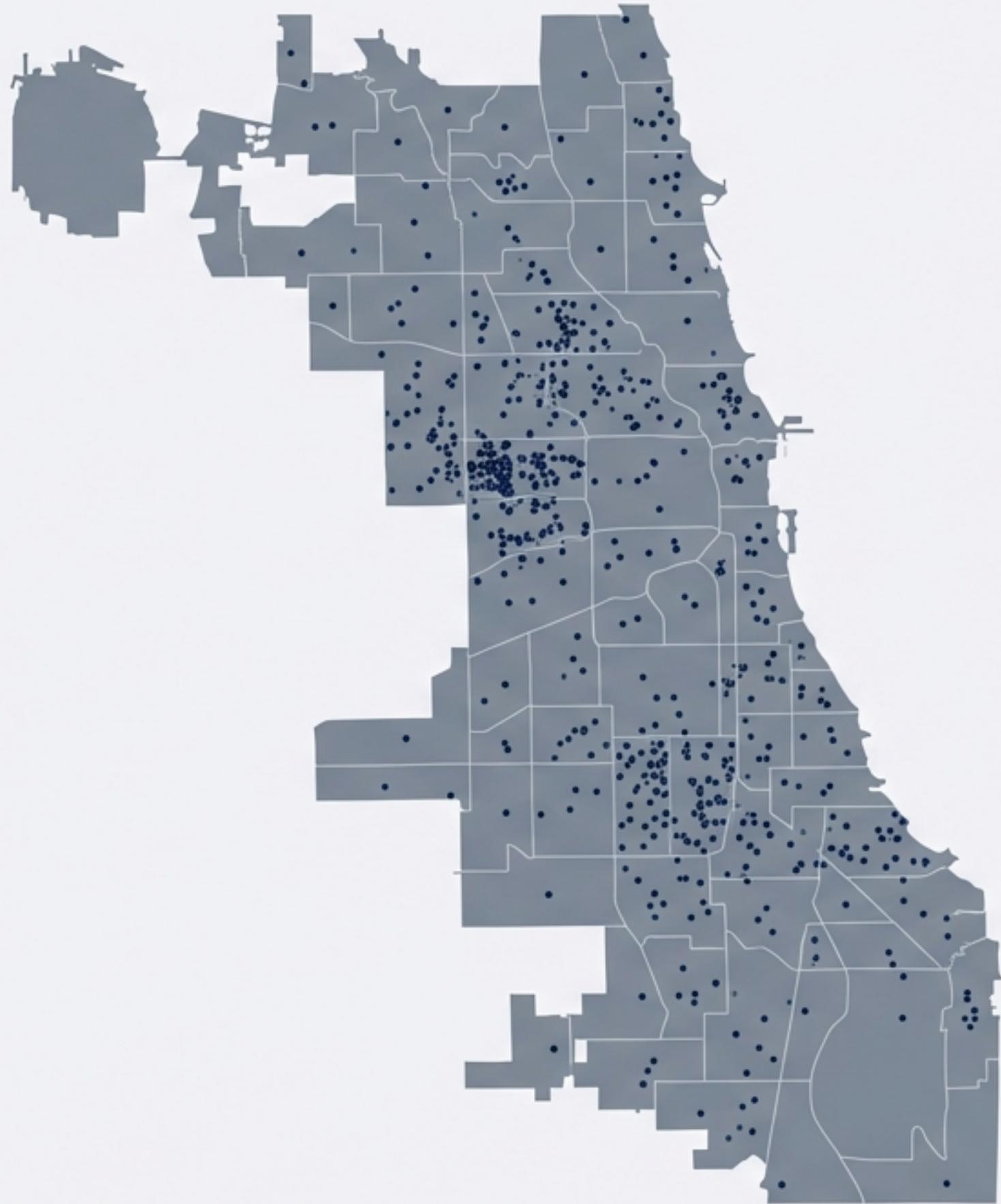
# Chicago Crime Analysis (2020-2024)

From Raw Data to Public  
Safety Strategy

**SCOPE:** 4-Year Incident Analysis

**GOAL:** Optimization of Resource Allocation

**TECH STACK:** Python, Pandas, Seaborn, Matplotlib



# The Ingestion Engine: Automating Data Retrieval

## The Approach

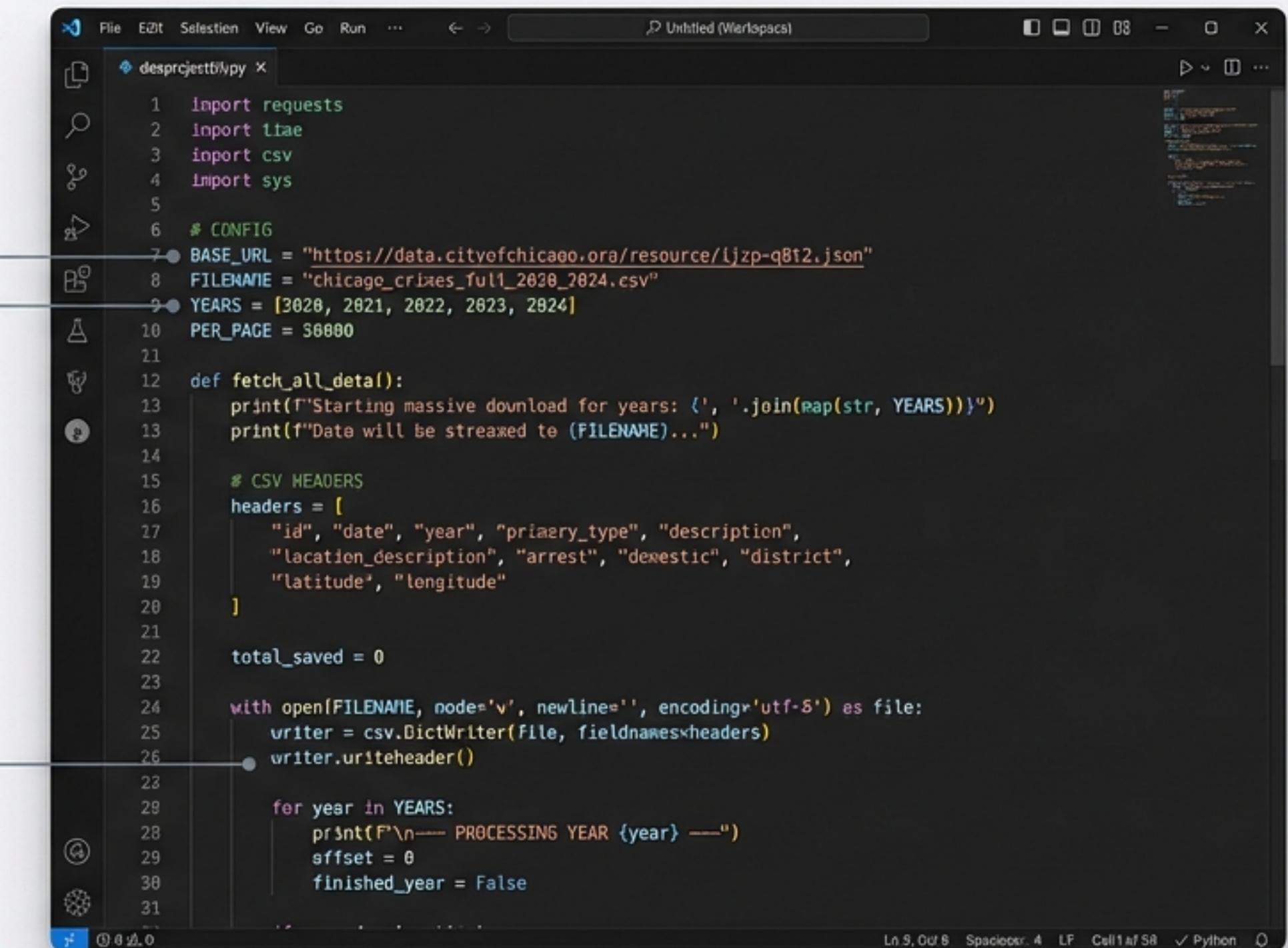
Custom Python script interfaces with the official City of Chicago API to retrieve data year-by-year.

## Handling Scale

Batch processing fetches 50,000 records per request with intentional delays to prevent server rejection.

## Robustness

Built-in error handling manages timeouts, saving a continuous stream to 'chicago\_crimes\_full\_2020\_2024.csv'.



A screenshot of a Jupyter Notebook interface showing a code cell with the following Python script:

```
File Edit Selection View Go Run ... Untitled (Workspaces) □□□□□ X
desprojectbipy.py
1 import requests
2 import time
3 import csv
4 import sys
5
6 # CONFIG
7 BASE_URL = "https://data.cityofchicago.org/resource/ljzp-q8t2.json"
8 FILENAME = "chicago_crimes_full_2020_2024.csv"
9 YEARS = [2020, 2021, 2022, 2023, 2024]
10 PER_PAGE = 50000
11
12 def fetch_all_data():
13     print(f"Starting massive download for years: {', '.join(map(str, YEARS))}")
14     print(f"Data will be streamed to {FILENAME}...")
15
16     # CSV HEADERS
17     headers = [
18         "id", "date", "year", "primary_type", "description",
19         "location_description", "arrest", "domestic", "district",
20         "latitude", "longitude"
21     ]
22
23     total_saved = 0
24
25     with open(FILENAME, mode='w', newline='', encoding='utf-8') as file:
26         writer = csv.DictWriter(file, fieldnames=headers)
27         writer.writeheader()
28
29         for year in YEARS:
30             print(f"\n--- PROCESSING YEAR {year} ---")
31             offset = 0
32             finished_year = False
33
34             while not finished_year:
35                 # Fetch data from API
36                 # ...
37
38                 # Write to CSV
39                 # ...
40
41                 # Check if all data for the year is processed
42                 # ...
43
44                 # Increment offset and check for finished year
45                 # ...
46
47             total_saved += 50000 * len(YEARS)
48
49     print(f"Total records saved: {total_saved}")
50
51
52
53
54
55
56
57
58
59
59
```

The code defines a `fetch_all_data` function that prints the years being processed and the output file name. It then sets up CSV headers and initializes a writer. The main loop iterates over each year, printing a header and then processing data in batches of 50,000 records until all data for the year is processed. Finally, it prints the total number of records saved.

# Initial Data Assessment: Identifying the Gaps

## DataFrame Info Output

#	Column	Non-Null Count	Dtype
0	id	1184260 non-null	int64
1	date	1184260 non-null	object
2	year	1184260 non-null	int64
3	primary_type	1184260 non-null	object
4	description	1184260 non-null	object
5	location_description	1177570 non-null	object
6	arrest	156951 non-null	object
7	domestic	233826 non-null	object
8	district	1184260 non-null	int64
9	latitude	1164531 non-null	float64
10	longitude	1164531 non-null	float64

dtypes: float64(2), int64(3), object(6)

memory usage: 99.4+ MB

## ⚠ The Challenge: Significant Missing Data

- Total Entries: 1,184,260
- Arrest: 1,027,309 null values (Critical Gap)
- Domestic: 950,434 null values
- Geospatial: 6,690 rows missing Latitude/Longitude

Insight: Analysis cannot proceed without addressing structural voids in 'Arrest' and 'Domestic' fields.

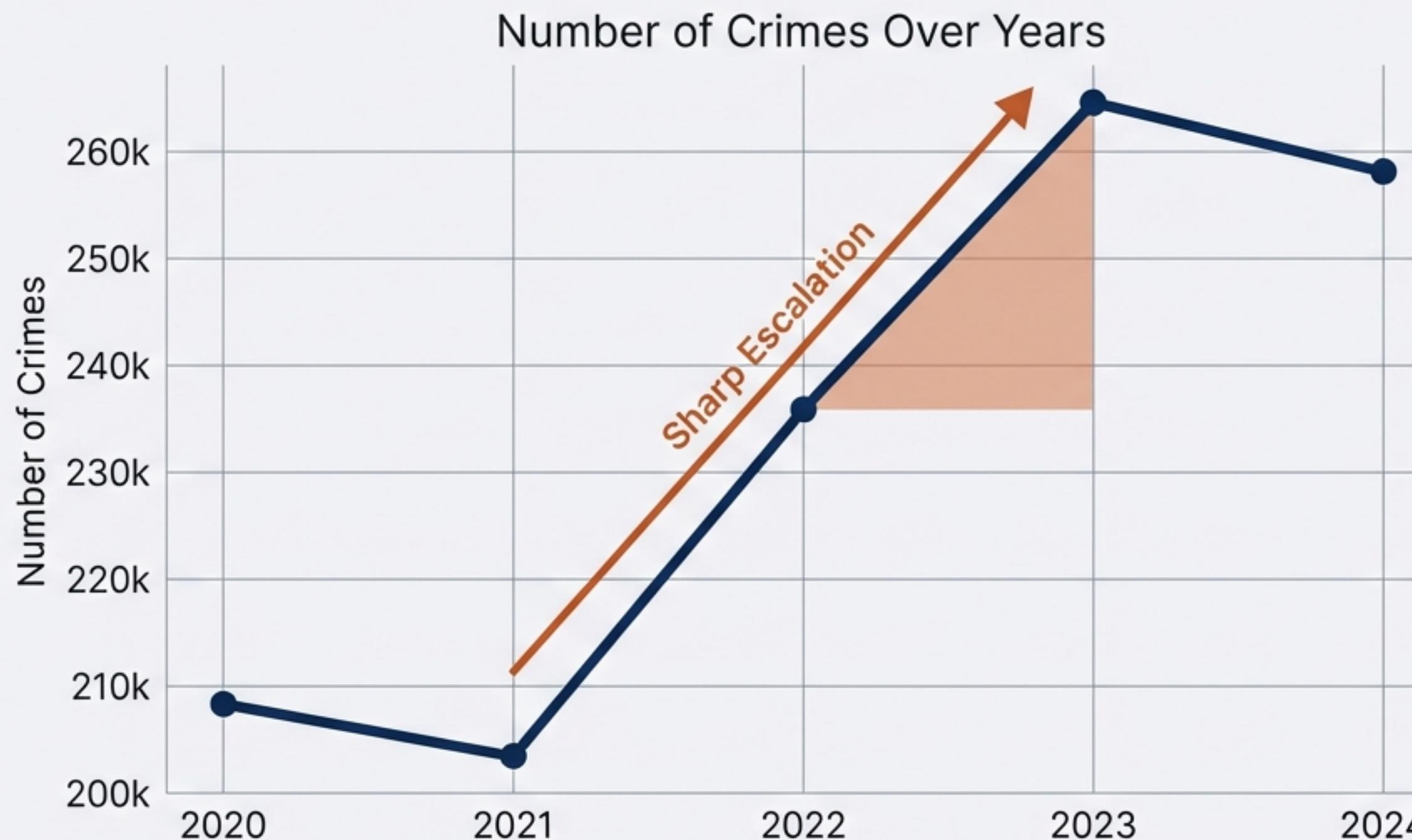
```
Column latitude:  
Number of null values: 6890  
  
Column arrest:  
Number of null values: 1027309  
  
Column domestic:  
Number of null values: 950434  
  
Column district:  
Number of null values: 0  
  
Column latitude:  
Number of null values: 19729  
  
Column longitude:  
Number of null values: 19729  
  
Column longitude:  
Number of null values: 19729
```

# The Cleaning Protocol: Standardization & Imputation



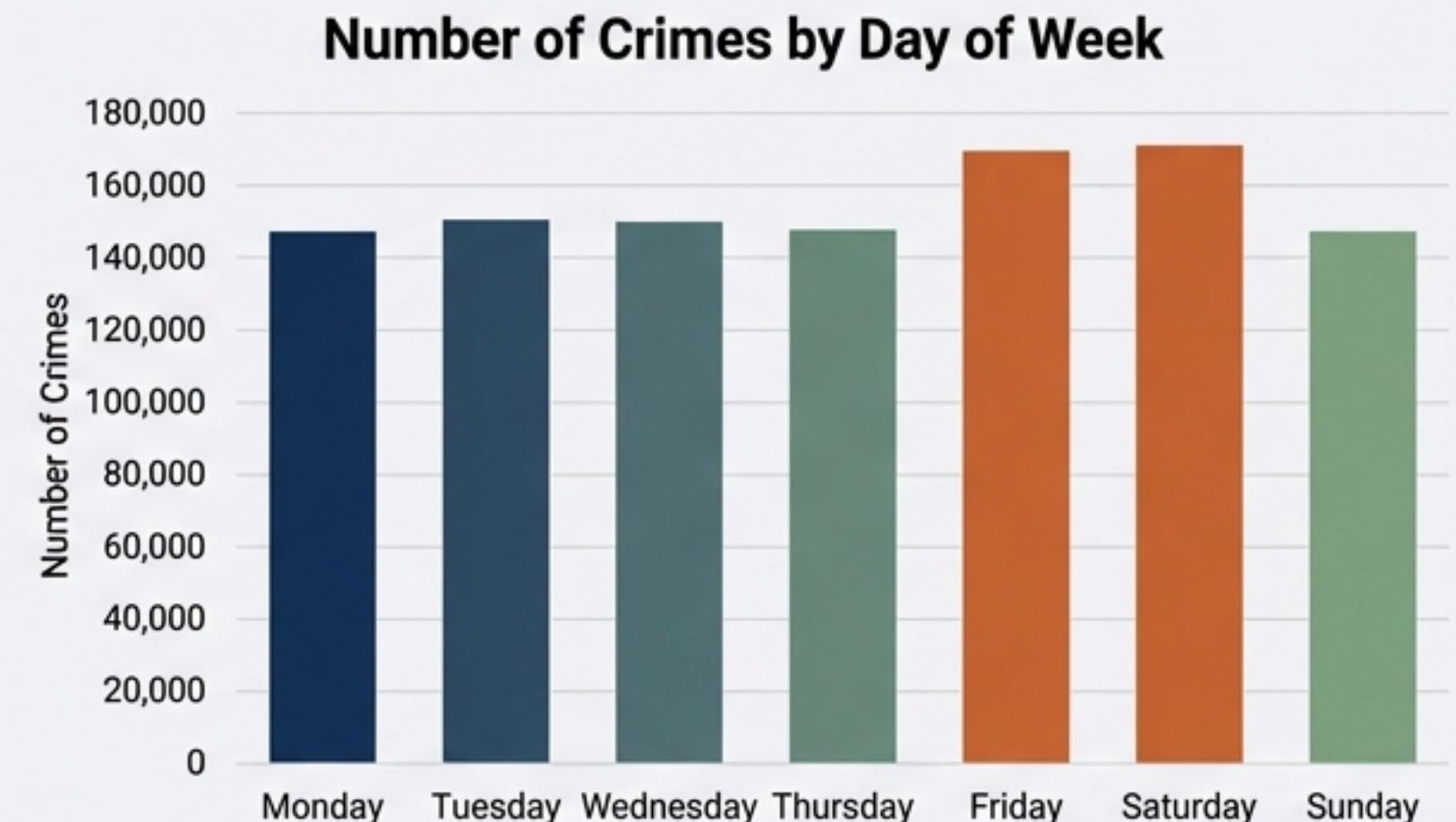
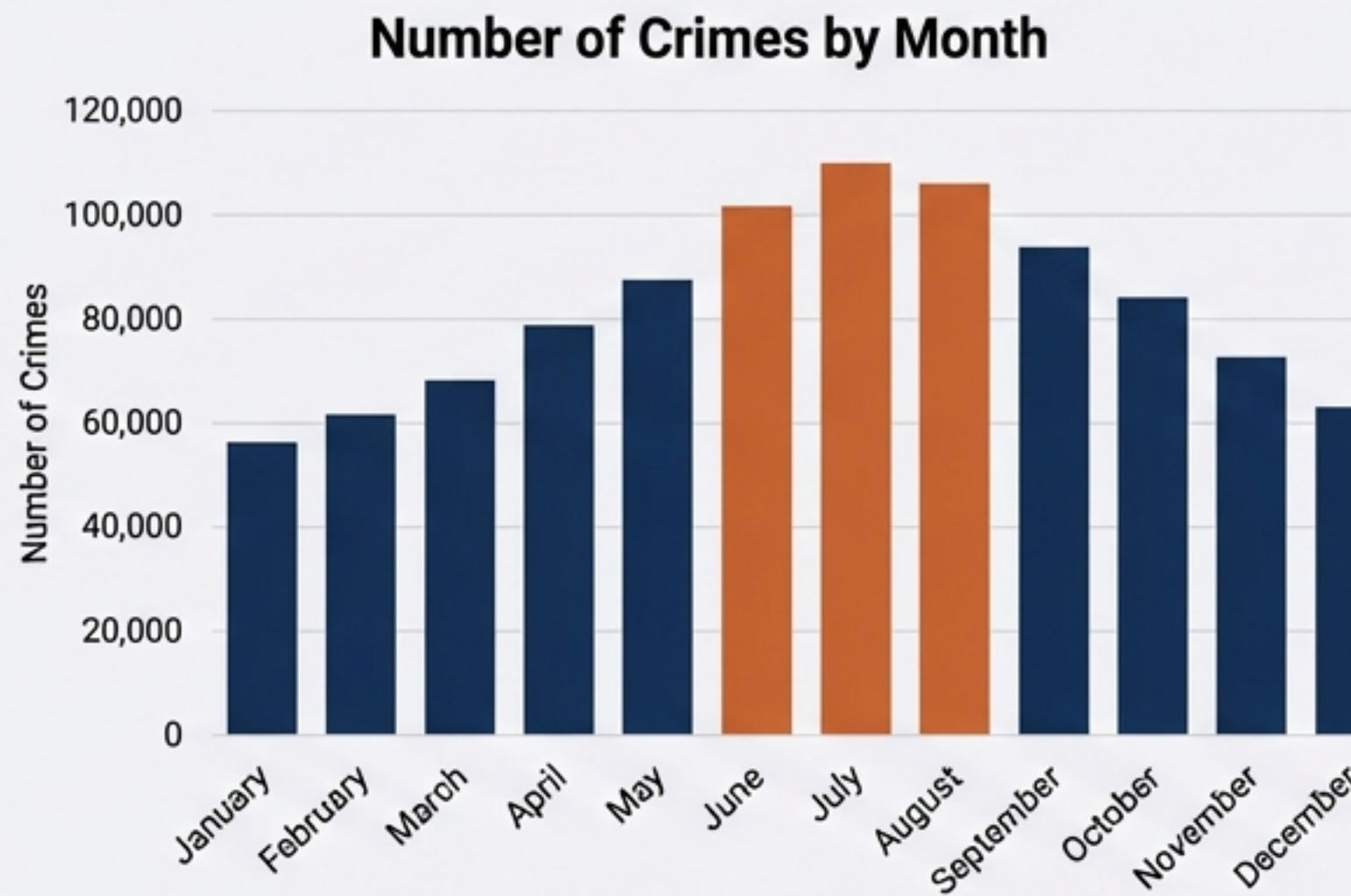
Result: Dataset is now standardized and ready for temporal and spatial analysis.

# Macro Trends: The Post-2021 Surge



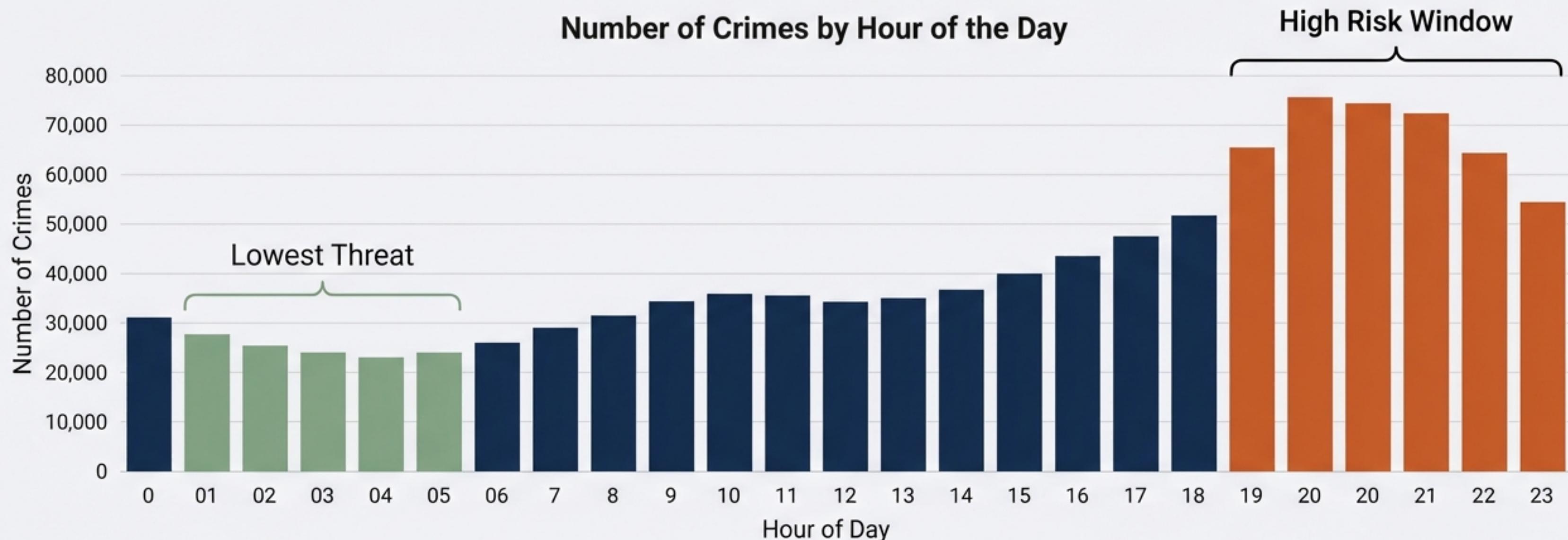
Observation: Crime rates hit a low in 2021 before a rapid surge, peaking in 2023. This suggests external social factors or policy shifts post-2021 have fundamentally altered incident volume.

# Seasonal Cyclicity: The Summer Peak



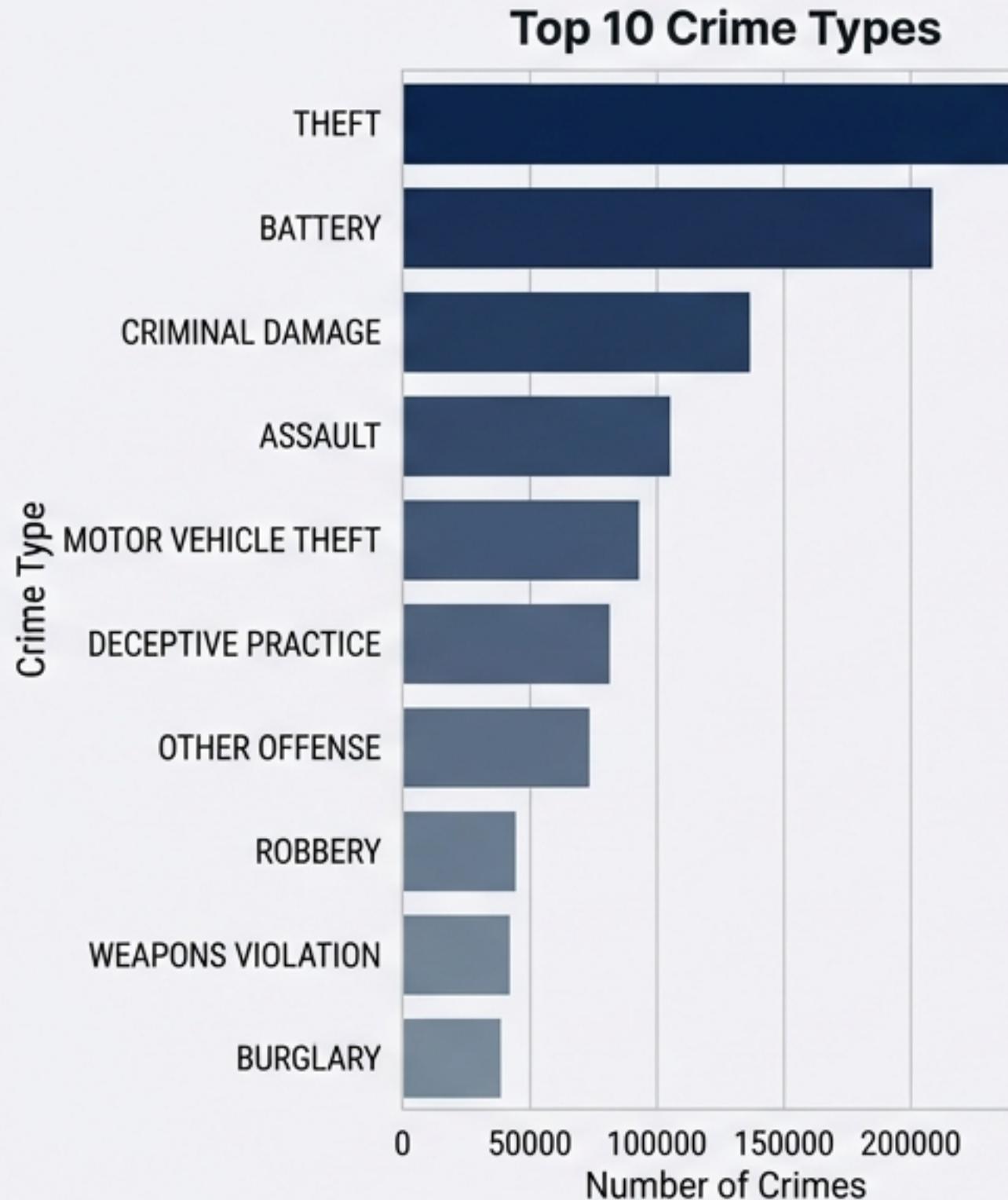
**Pattern Analysis:** Activity follows the thermometer. The 'Summer Strategy' requires peak resource deployment in June-August and on weekends.

# The 24-Hour Curve: Evening Vulnerability



**Tactical Response:** Shift changes must avoid the 6 PM–8 PM window. Public lighting initiatives should target this specific evening timeframe.

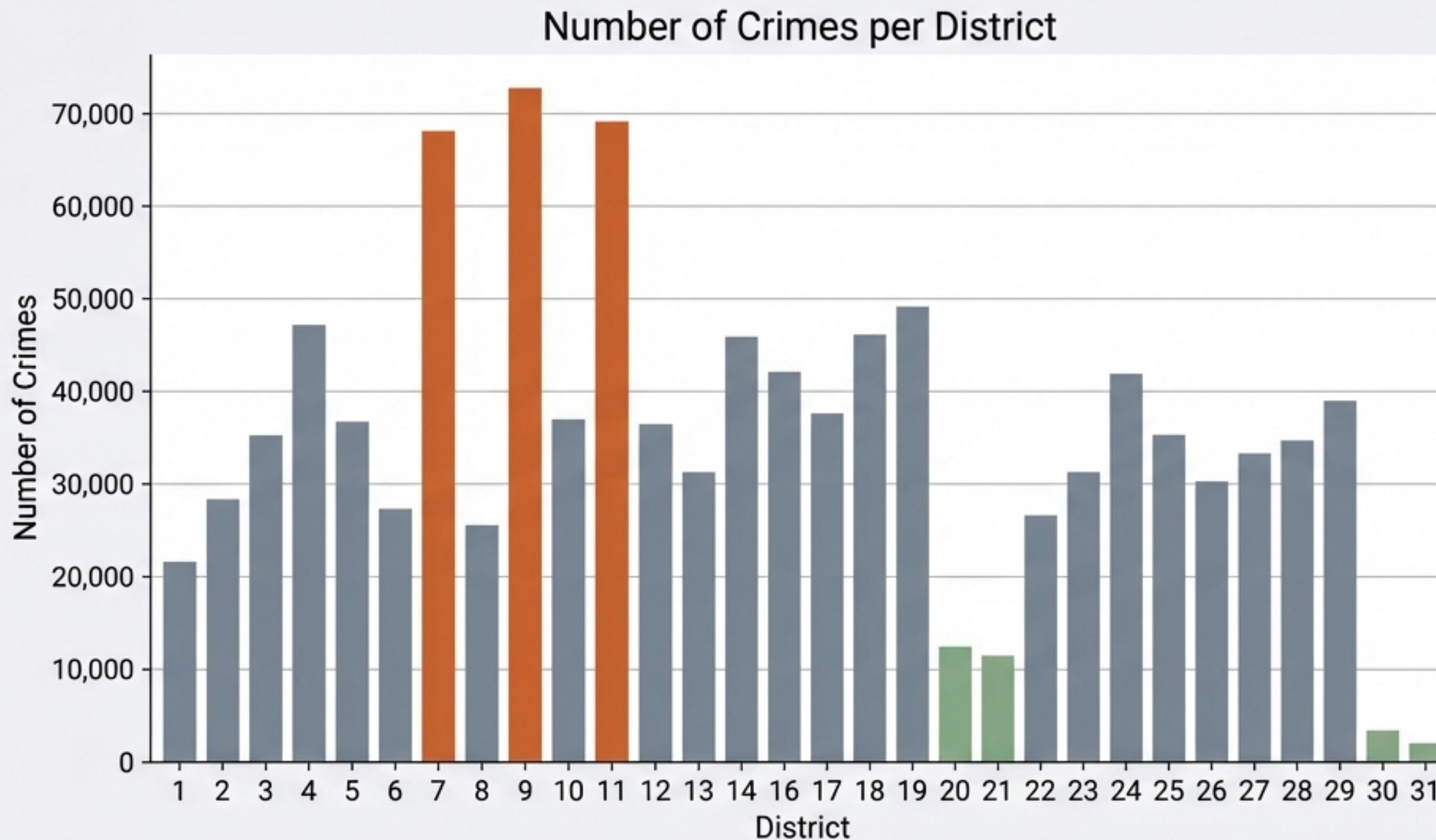
# Volume Drivers: Theft and Battery Dominate



**Key Insight:** Property crime (Theft) and policing private spaces (Apartments/Residences) are the primary volume challenges, outweighing violent street crime in pure numbers.



# Geographic Disparity: The Burden on Districts 6 & 8

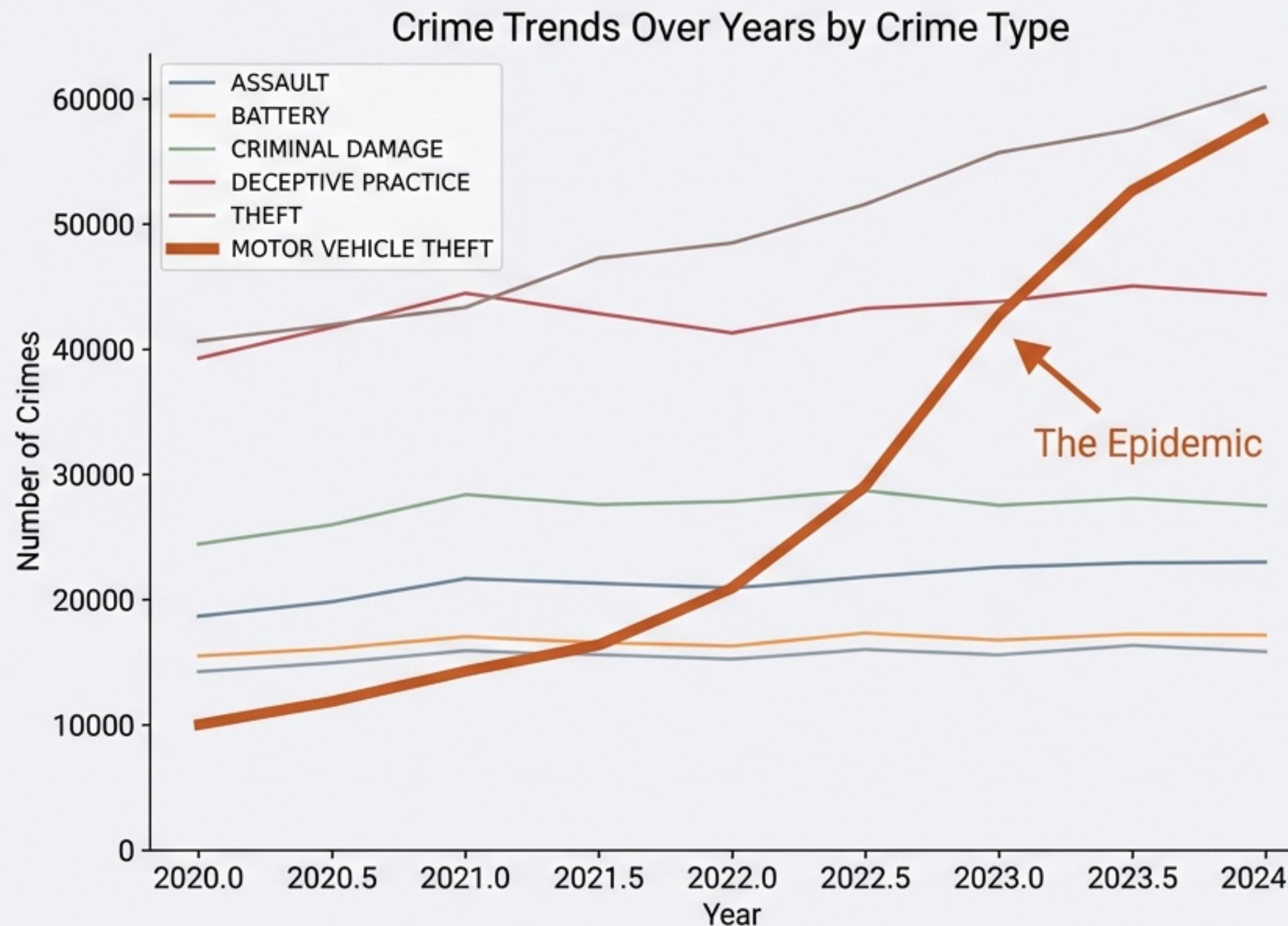


Uneven Distribution:  
Safety is not  
geographically equal.

- Hotspots: Districts 6, 8, 11
- Low Activity: Districts 20, 31

Recommendation: End  
flat resource  
distribution. Personnel  
must be weighted  
heavily toward the  
orange zones.

# Emerging Threats: The Motor Vehicle Theft Anomaly

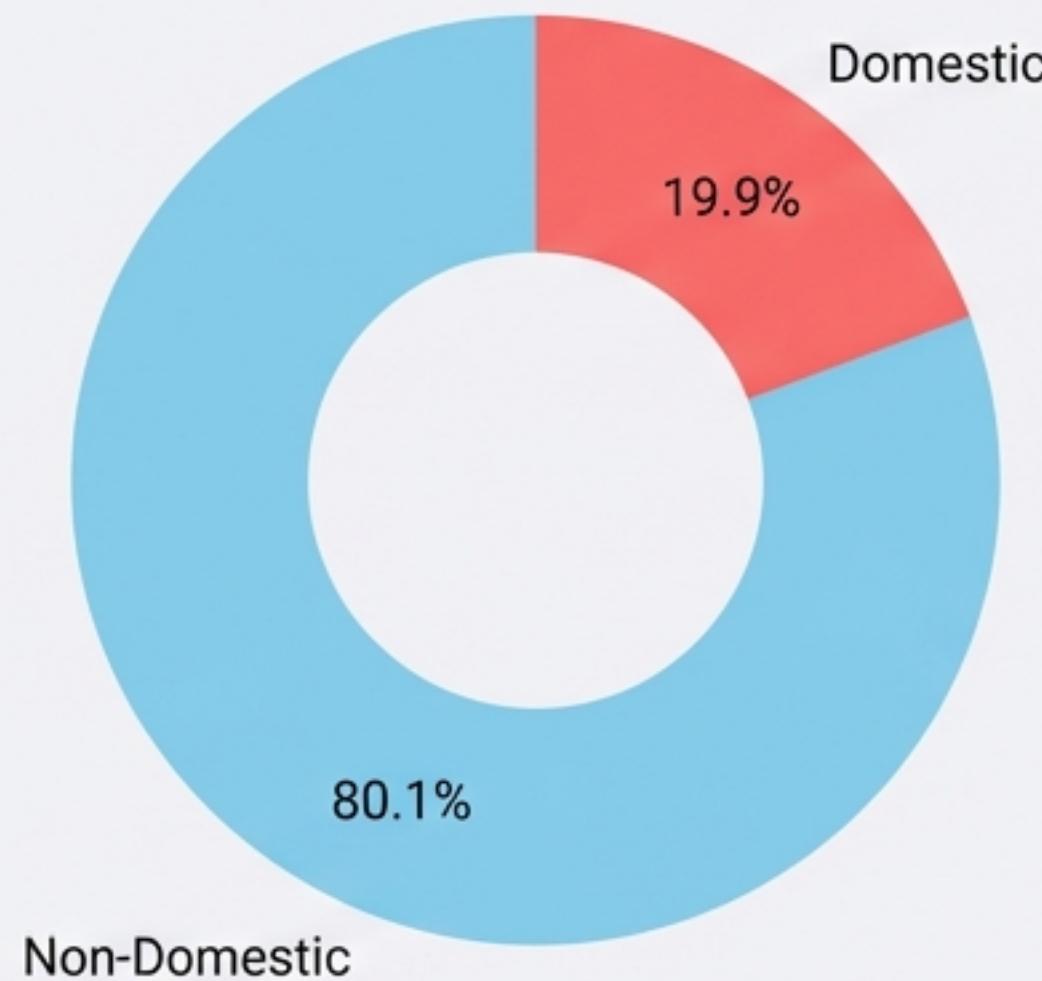


While other crime types remain stable, Motor Vehicle vehicle Theft exhibits a drastic divergence starting late 2021.

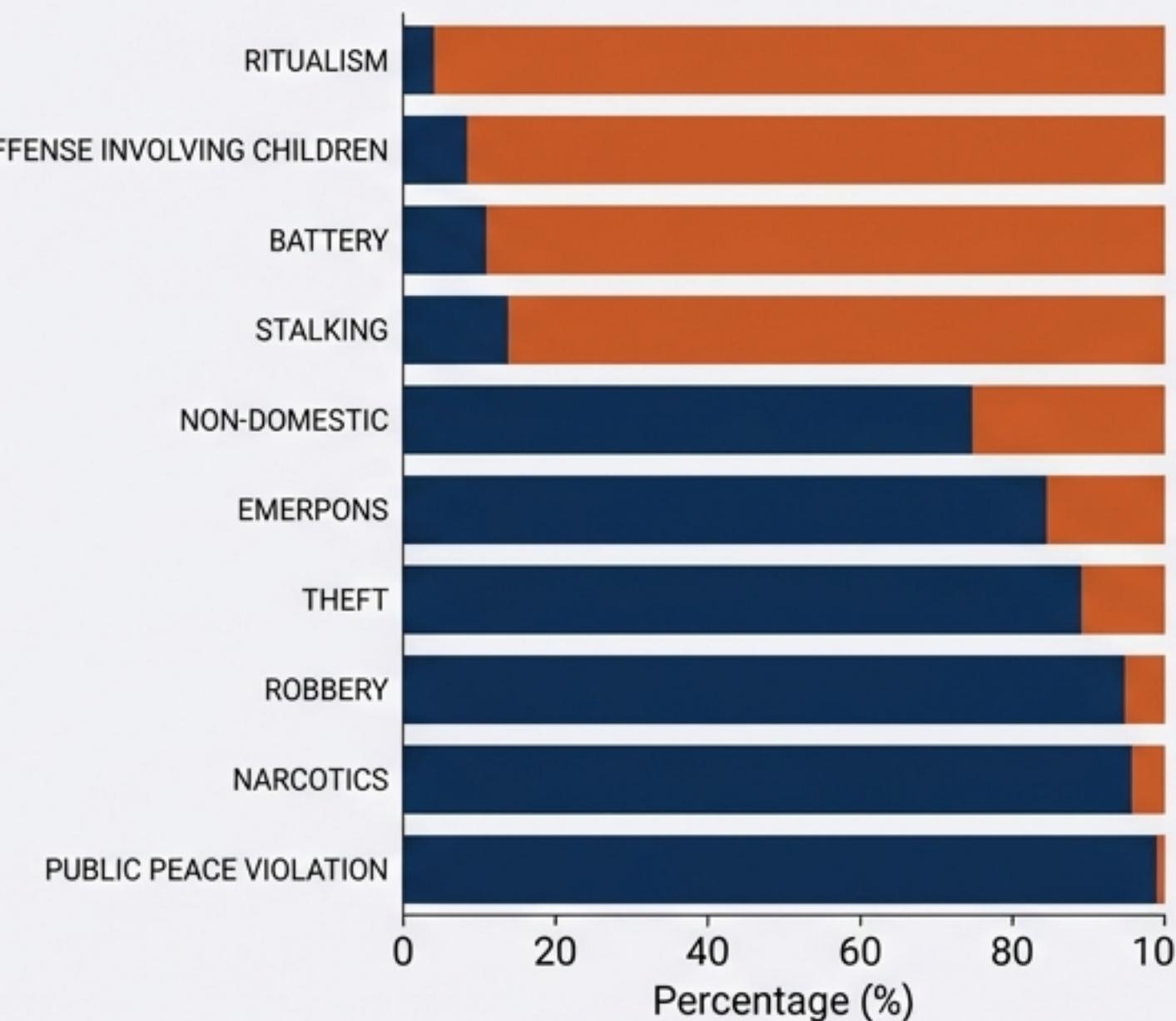
This specific anomaly requires a dedicated task force and technology-based intervention (LPRs), as standard tactics are failing.

# The Social Dimension: 20% Domestic Involvement

Domestic vs Non-Domestic Crimes

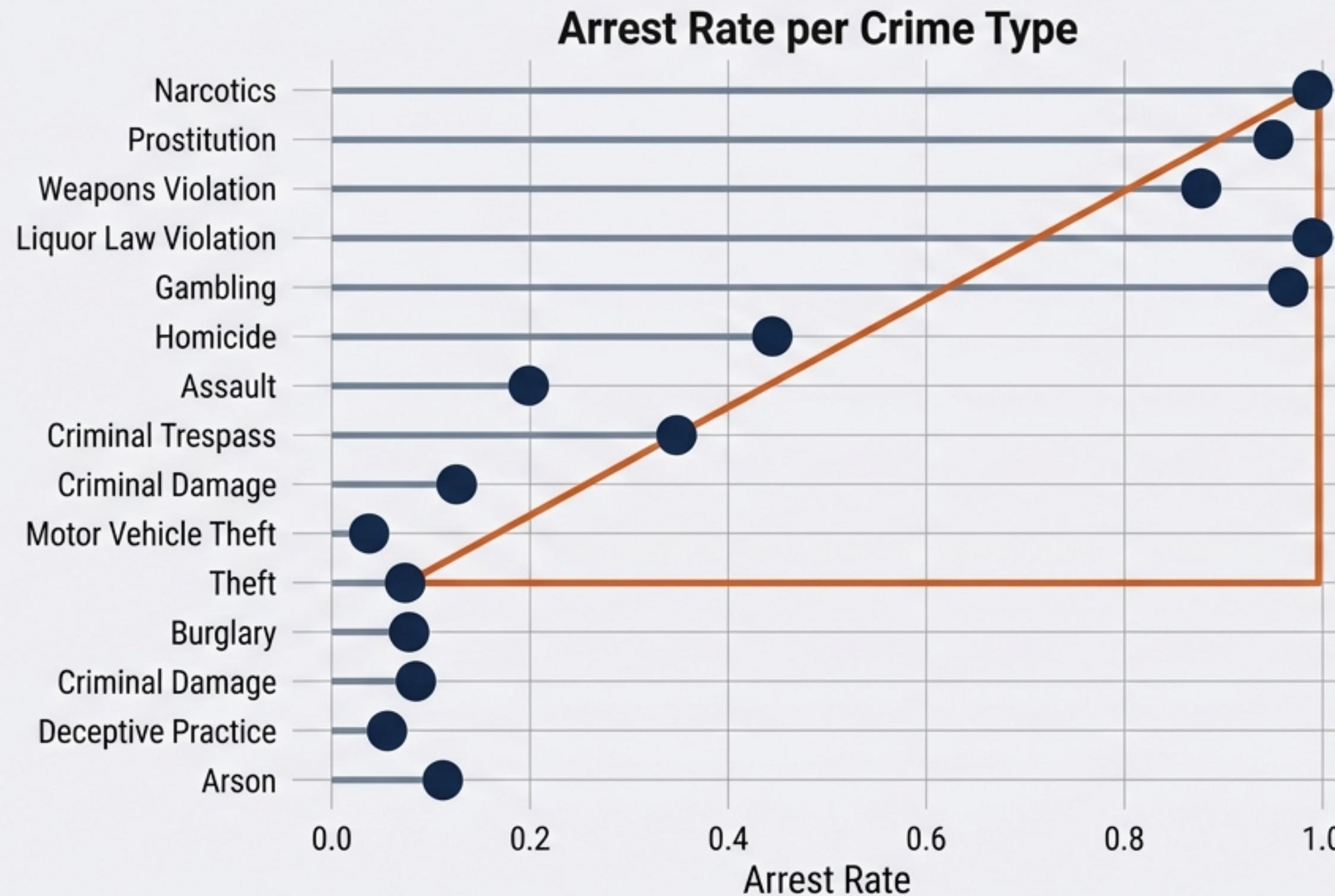


Percentage of Domestic vs Non-Domestic Crimes by Crime Type



1 in 5 crimes requires **social intervention**, not just policing. Crimes involving children are almost exclusively domestic, demanding **integrated social work teams**.

# The Accountability Gap: Arrest Rates by Offense

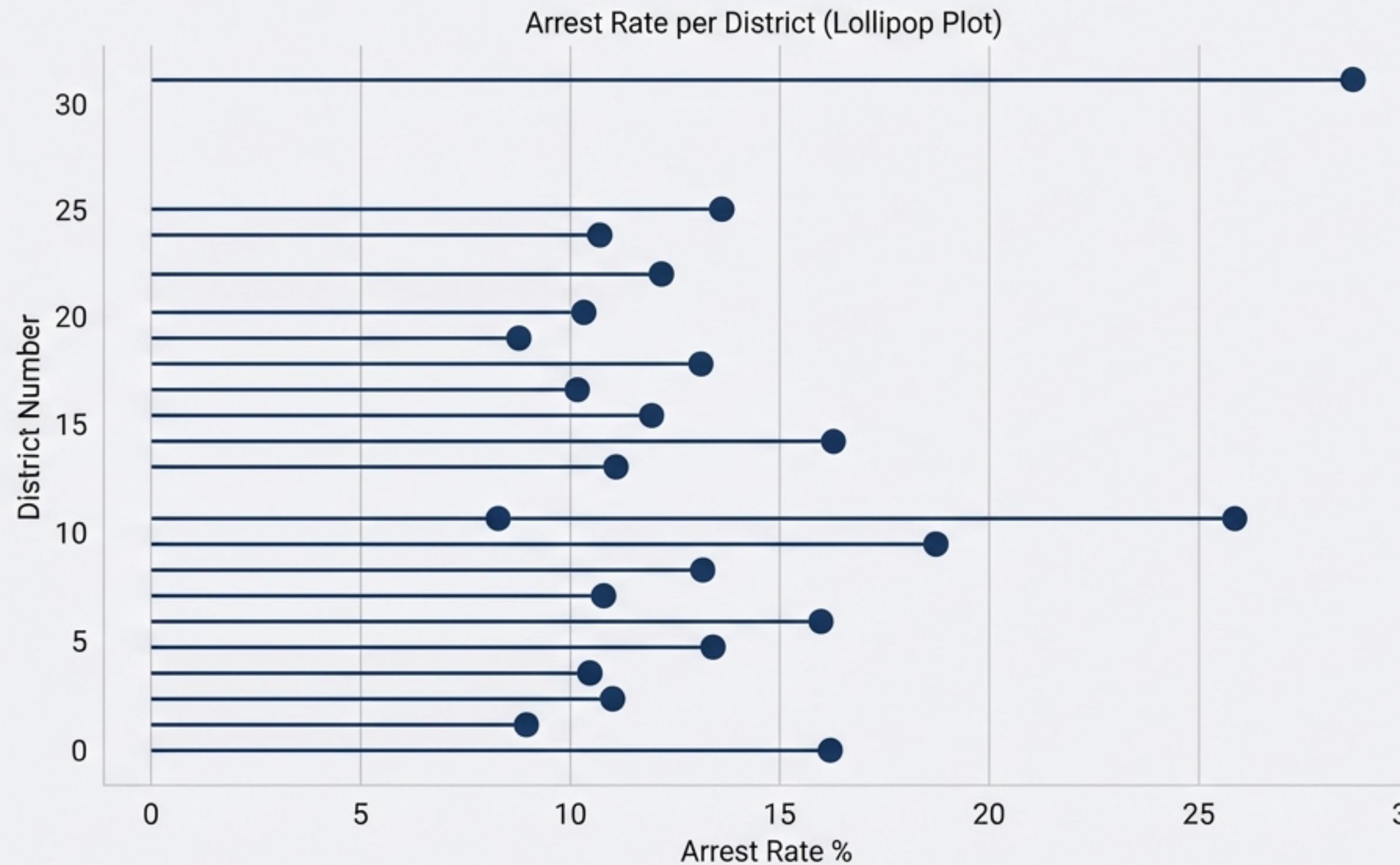


## The Impunity Problem

High-volume crimes (**Theft, Motor Vehicle Theft**) have arrest rates <10%.

The system is effective at vice crimes but struggles with property accountability.

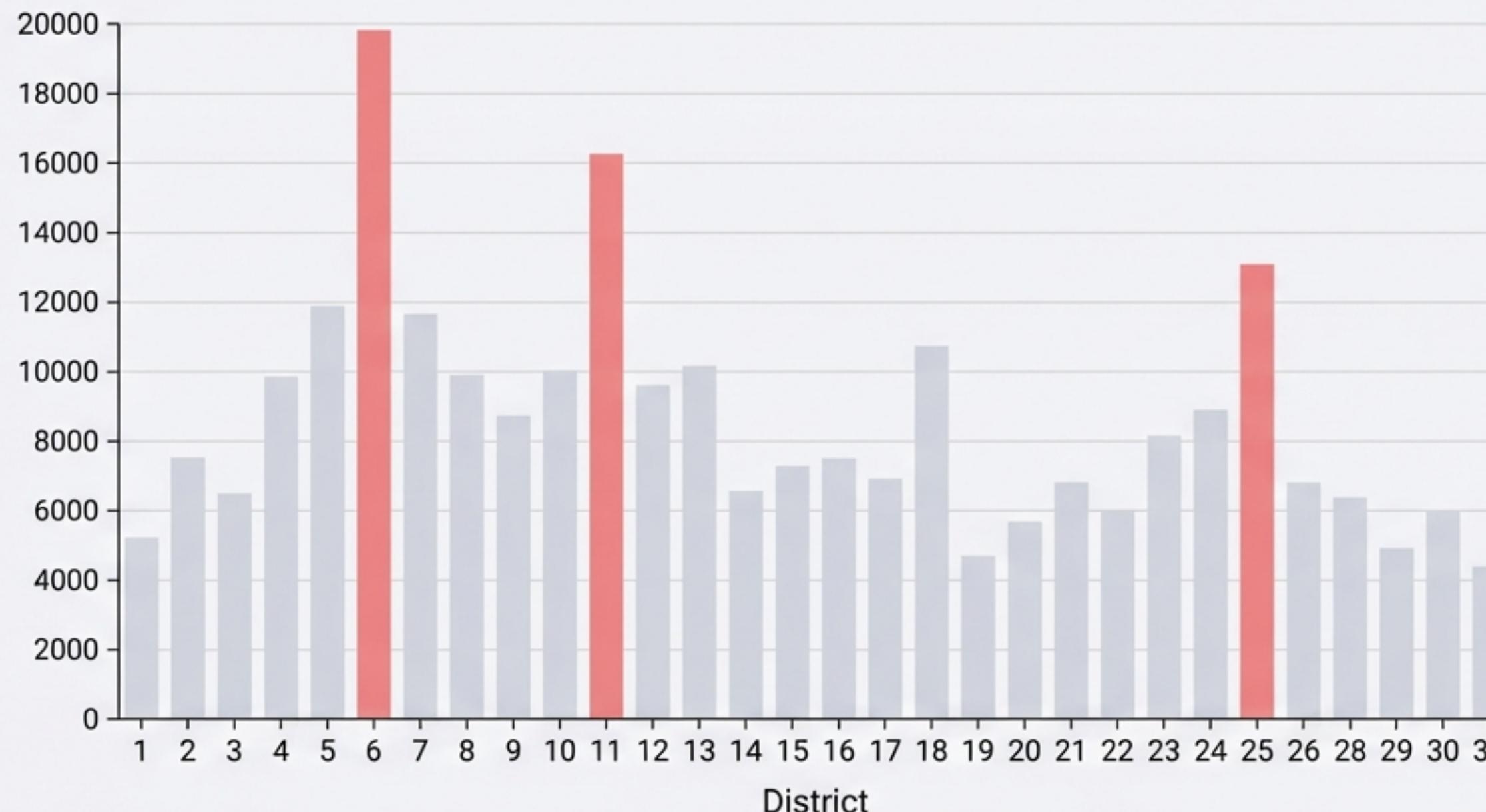
# District Performance: Variance in Enforcement



Performance disparity is evident. Low-performing districts need audits to determine if the bottleneck is manpower, resources, or the specific nature of local crime types.

# Targeting Support: Domestic Crime Hotspots

**Number of Domestic Crimes by District**



## Social Resource Deployment:

Family protection services and de-escalation training must be aggressively funneled into Districts 6, 11, and 25.

A city-wide approach will dilute impact; a targeted approach addresses the root cause.

# Strategic Roadmap: Data-Driven Interventions



## TEMPORAL DEPLOYMENT

Implement 'Summer Surge' protocols (June-Aug) and overlapping shifts during the 6 PM-11 PM peak window.



## TARGETED ENFORCEMENT

Create a specialized task force for Motor Vehicle Theft and focus investigative resources on Theft to improve the <10% arrest rate.



## GEOSPATIAL PRIORITIZATION

End flat resourcing. Shift heavy patrol weight to Districts 6, 8, and 11.



## SOCIAL INTEGRATION

Deploy family protection services specifically to District 25 and 6 to handle the 20% domestic volume.

**Data is the roadmap to a safer Chicago.**