NYPD Crime Overview

Ali Adnan, Ayushman Shrestha, Hongyan, Shreya Karki

Data Cleaning Process

Initial Dataset
8 million+ crime records

→

Timeframe Filter
Focus on crimes from 2013 to 2023

→

Cleaning Criteria
Remove rows with >3 null values

→

Eliminate Invalid Data
Removed invalid values

→

Result
Dataset reduced to 3.4 million records

SUSP_AGE_GROUP	
UNKNOWN	1262252
25-44	1220372
45-64	438550
18-24	424517
<18	129064
...	...
967	1
964	1
-976	1
-5	1
-928	1

Imputing and Standardizing Values

Standardized gender categories

VIC_SEX	
F	1476652
M	1149744
D	468929
E	315043
L	5885

VIC_SEX	count
F	1476652
M	1149744
UNKNOWN	783972
L	5885

Filled missing values with the mode

```
#find  unique combination of borough and crime category
for boro in filtered_df['BORO_NM'].dropna().unique():  # Skip missing borough names
    for crime in filtered_df['CRIME_CTGORY'].dropna().unique():  # Skip missing crime types

        # condition for current borough and crime
        condition = (filtered_df['BORO_NM'] == boro) & (filtered_df['CRIME_CTGORY'] == crime)

        # Find the most common value
        most_common_age_group = filtered_df.loc[condition, 'SUSP_AGE_GROUP'].mode()

        # If there's at least one common value, fill missing values
        if not most_common_age_group.empty:
            filtered_df.loc[condition & filtered_df['SUSP_AGE_GROUP'].isna(), 'SUSP_AGE_GROUP'] = most_common_age_group.iloc[0]
        else:
            # fill with Unknown if no mode is found
            filtered_df.loc[condition & df_copy['SUSP_AGE_GROUP'].isna(), 'SUSP_AGE_GROUP'] = 'UNKNOWN'
```

Statistical Analysis- Descriptive

CRIME_STATUS YEAR	COMPLETED	ATTEMPTED	Total	Completed %	Attempted %
2013	111471	1745	113216	98.46	1.54
2014	120995	1898	122893	98.46	1.54
2015	202804	3892	206696	98.12	1.88
2016	337596	6380	343976	98.15	1.85
2017	338567	6277	344844	98.18	1.82
2018	338627	6005	344632	98.26	1.74
2019	344674	6075	350749	98.27	1.73
2020	313499	5346	318845	98.32	1.68
2021	355376	5718	361094	98.42	1.58
2022	428724	6719	435443	98.46	1.54
2023	466851	7013	473864	98.52	1.48

Status of Crime

CRIME_CTGORY	count
MISDEMEANOR	1714135
FELONY	1020612
VIOLATION	681506

CRIME_CTGORY	2013	2014	2015	2016	2017	2018	2019	2020	2021	2022	2023
FELONY	26279	29232	59171	98939	96701	96854	101920	99516	114640	141849	155511
MISDEMEANOR	49539	52890	101725	181902	184839	181037	180574	155758	174624	213785	237462
VIOLATION	37398	40772	45800	63135	63304	66741	68255	63571	71830	79809	80891

Crime Trend (2013-2023)

Hourly Crime Distribution Statistics:

	count	mean	std	min	25%	50%	75%	max	range	IQR
HOUR	3416253.0	13.171977	6.506642	0.0	9.0	14.0	18.0	23.0	23.0	9.0

Hourly Distribution

Statistical Analysis- Counts

SUSP_SEX	count
M	2085630
UNKNOWN	712516
F	618107

dtype: int64

Suspect and Victim

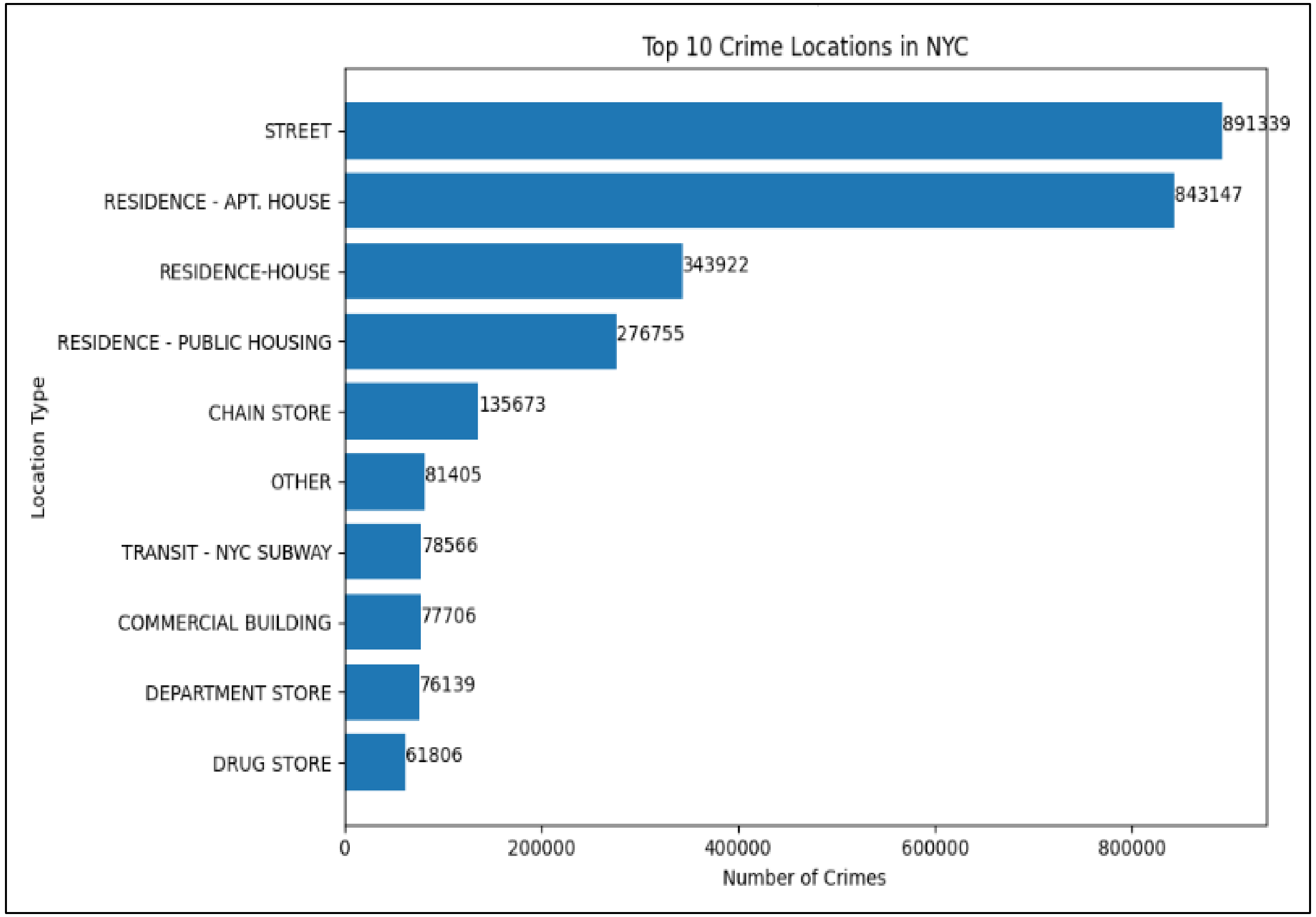
```
SUSP_AGE_GROUP
UNKNOWN     1262178
25-44       1172612
45-64        419571
18-24        399576
<18          122392
65+           39924
Name: count, dtype: int64
1262178
```

Suspect Age

```
VIC_AGE_GROUP
25-44       1260985
UNKNOWN      851045
45-64        644048
18-24        357328
<18          164558
65+          138289
Name: count, dtype: int64
1260985
```

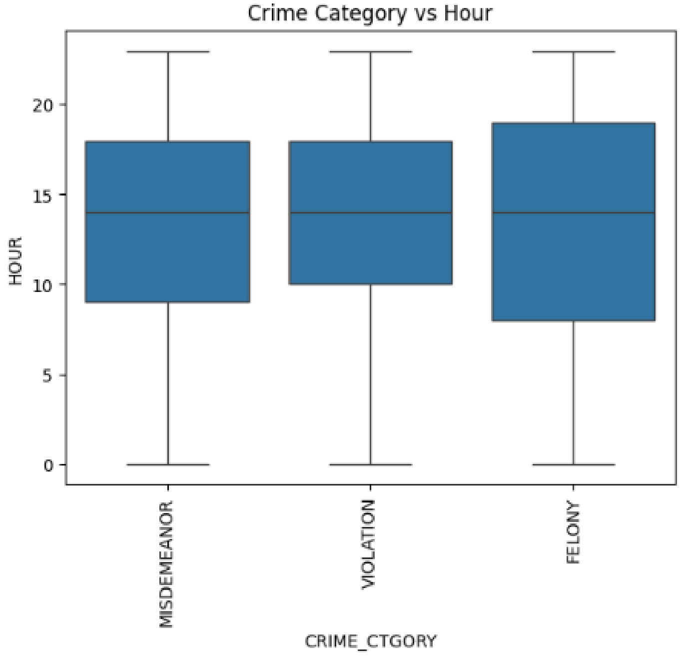
Victim Age

Bar Chart & Box Plot



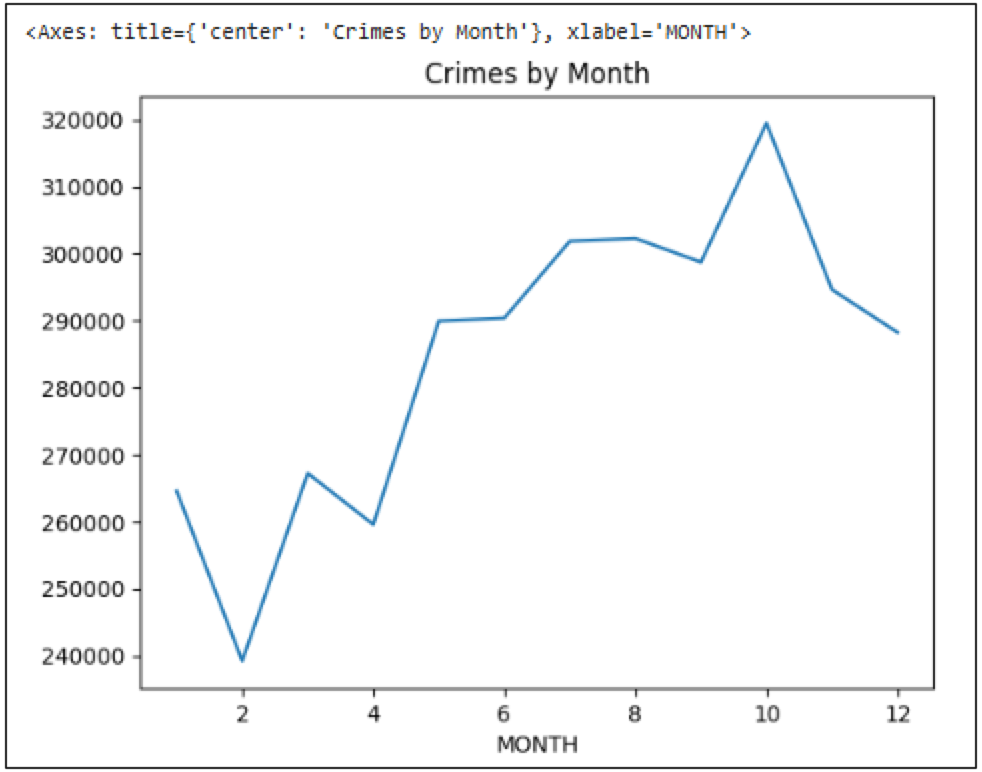

```python
import seaborn as sns

# Gender pairing
gender_pair = pd.crosstab(filtered_df['SUSP_SEX'], filtered_df['VIC_SEX'])
print("Suspect-Victim Gender Crosstab:")
print(gender_pair)

# Crime category by hour
sns.boxplot(x='CRIME_CTGORY', y='HOUR', data=filtered_df)
plt.xticks(rotation=90)
plt.title('Crime Category vs Hour')
plt.show()
```

```
Suspect-Victim Gender Crosstab:
VIC_SEX        F     L       M  UNKNOWN
SUSP_SEX
F         324432   868  193549    99258
M         889165  4573  623995   567897
UNKNOWN   263055   444  332200   116817
```

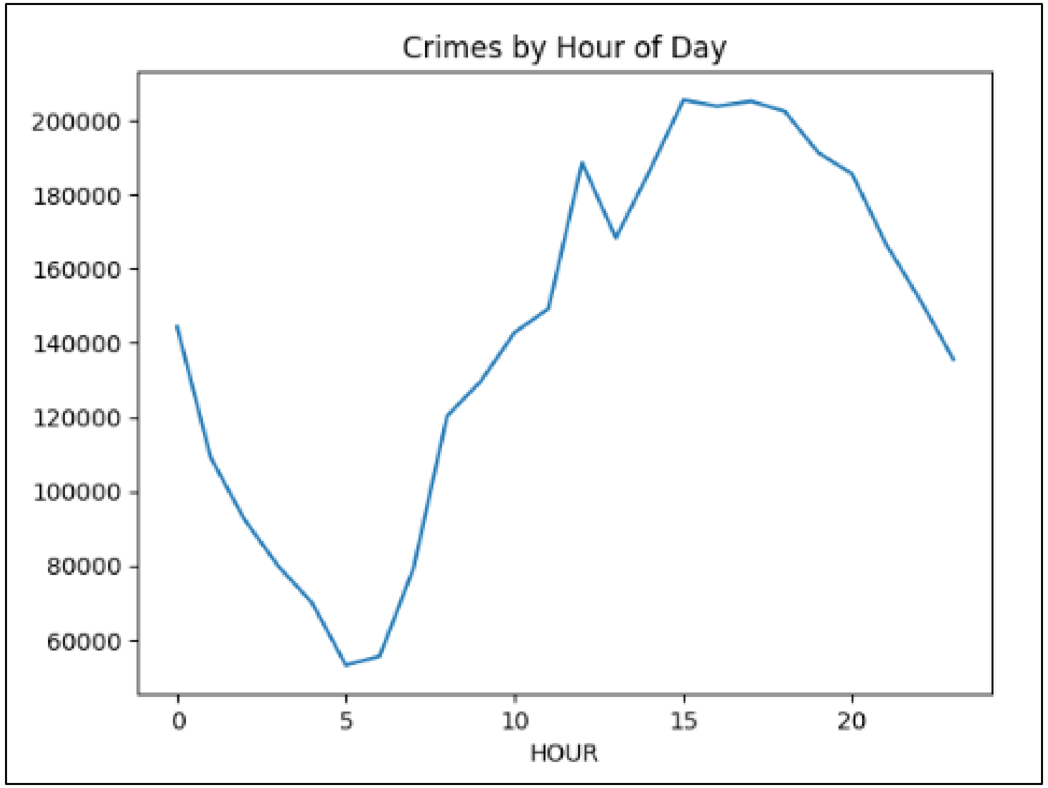


Crime Timing and Gender Patterns

Line Plot



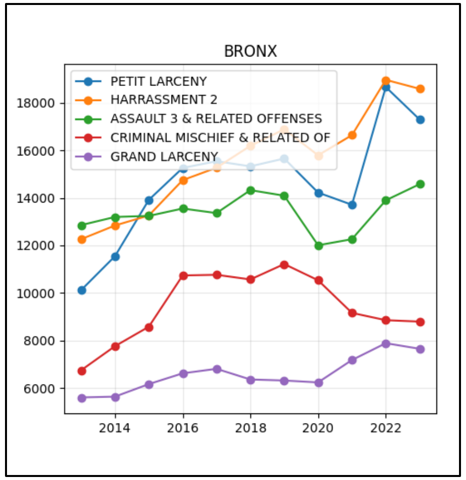
Crimes per month

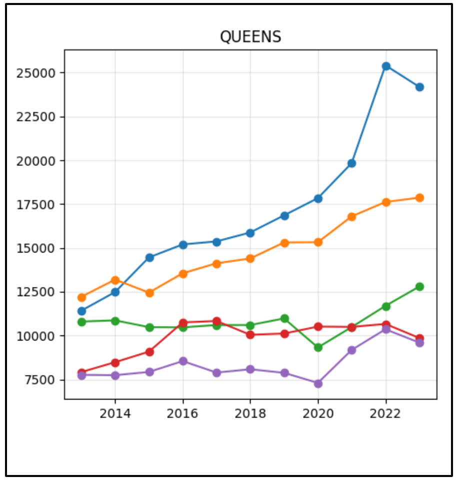


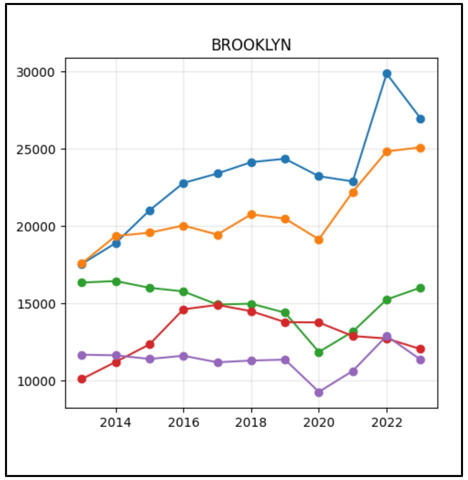
Crimes by hour

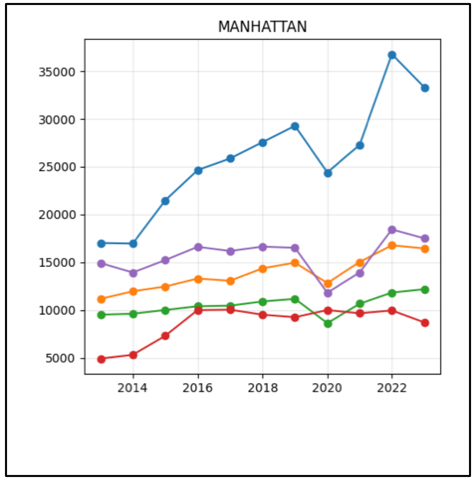
Line Charts Representing Boroughs

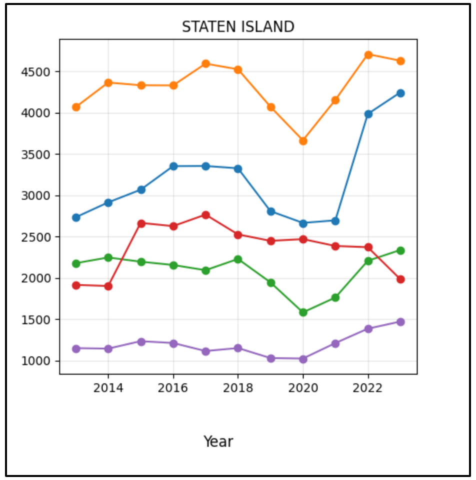
Top 5 crimes from 2013 to 2023





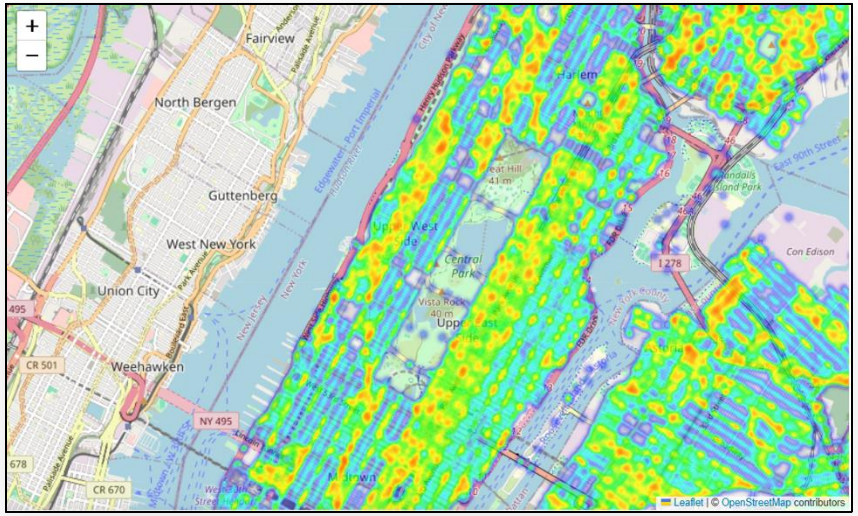


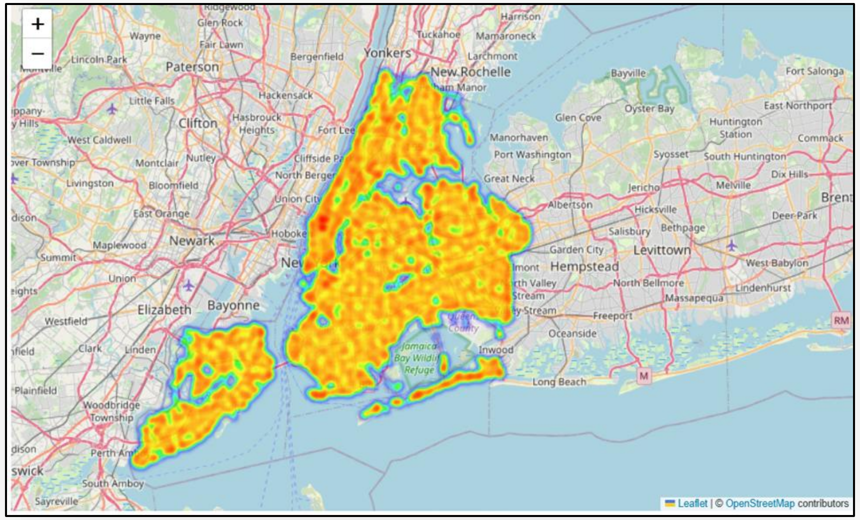




NYC Heatmap

```
import folium
from folium.plugins import HeatMap

df_map = filtered_df.dropna(subset=['LAT', 'LONG'])
nyc_map = folium.Map(location=[40.7128, -74.0060], zoom_start=11)

heatmap_data = df_map[['LAT', 'LONG']].values.tolist()

# Adjust Heatmap parameters
HeatMap(
    heatmap_data,
    radius=5,  # Decreased radius for more concentrated heat
    blur=4,    # Decreased blur for sharper heat spots
    max_zoom=18 # Increased max_zoom for better visibility at higher zoom lev
).add_to(nyc_map)

nyc_map
```





Prediction Model

	precision	recall	f1-score	support
ADMINISTRATIVE CODE	0.76	0.13	0.23	1841
ASSAULT 3 & RELATED OFFENSES	0.57	0.88	0.69	94462
BURGLARY	0.55	0.42	0.48	15266
CRIMINAL MISCHIEF & RELATED OF	0.53	0.18	0.26	47906
CRIMINAL TRESPASS	0.40	0.10	0.16	4156
DANGEROUS DRUGS	0.48	0.75	0.59	18889
DANGEROUS WEAPONS	0.41	0.22	0.29	8924
FELONY ASSAULT	0.48	0.61	0.54	39203
FORGERY	0.47	0.54	0.50	6416
FRAUDS	0.48	0.05	0.10	2309
GRAND LARCENY	0.58	0.71	0.64	46690
GRAND LARCENY OF MOTOR VEHICLE	0.37	0.29	0.32	7740
HARRASSMENT 2	1.00	1.00	1.00	135053
INTOXICATED & IMPAIRED DRIVING	0.59	0.68	0.63	5401
MISCELLANEOUS PENAL LAW	0.46	0.46	0.46	24821
NYS LAWS-UNCLASSIFIED FELONY	0.78	0.34	0.47	970
OFF. AGNST PUB ORD SENSBLTY &	0.46	0.24	0.32	36442
OFFENSES AGAINST PUBLIC ADMINI	0.48	0.08	0.13	12815
OFFENSES AGAINST THE PERSON	0.51	0.08	0.14	1913
OFFENSES INVOLVING FRAUD	0.46	0.16	0.23	2379
OTHER	0.59	0.23	0.33	3864
OTHER OFFENSES RELATED TO THEF	0.47	0.28	0.35	2048
PETIT LARCENY	0.74	0.82	0.78	104239
POSSESSION OF STOLEN PROPERTY	0.40	0.02	0.03	2259
RAPE	0.67	0.70	0.68	2710
ROBBERY	0.52	0.51	0.51	27751
SEX CRIMES	0.87	0.81	0.84	10412
THEFT-FRAUD	0.44	0.02	0.03	3096
UNAUTHORIZED USE OF A VEHICLE	0.69	0.13	0.21	2512
UNKNOWN	0.43	0.02	0.03	178
VEHICLE AND TRAFFIC LAWS	0.44	0.23	0.30	10586
accuracy			0.66	683251
macro avg	0.55	0.38	0.40	683251
weighted avg	0.65	0.66	0.63	683251