# Predictive Analysis of NYPD Complaint Data

Ilnaz Magizov
Alexey Shulmin
Aleksandr Skvorcov
Ilya Krasheninnikov

May 9, 2025

## Contents

# 1 Introduction

## 1.1 Problem Overview

The goal of this project is to predict the **level of offense** (crime category) for police complaints, given all other details of the incident. The levels of offense are **felony, misdemeanor**, and **violation**, as defined by the NYPD database used for training. The **NYPD Complaint Data Historic** dataset, containing approximately 6.5 million reported crime incidents in NYC from 2006 through 2023, was selected for its size and diversity, contributing to model efficiency.

## 1.2 Business Objectives

Digitalization improves efficiency across various societal spheres, including crime prevention. Our project aims to lay the foundation for a crime prediction and detection system that police can use to reduce crime rates.

# 2 Dataset Overview

## 2.1 Source & Schema

The primary dataset is downloaded from Kaggle. After concatenation, the staging schema contains **35 fields**, including:

- `CMPLNT_NUM` *(bigint)* — unique complaint ID.
- `CMPLNT_FR_DT` & `CMPLNT_FR_TM` — offence start date/time in Unix ms.
- `ADDR_PCT_CD`, `BORO_NM`, `X_COORD_CD`, `LATITUDE`.
- `KY_CD`, `PD_DESC` — legal offence keys.
- `LAW_CAT_CD` *(target)* — Felony, Misdemeanor, Violation.
- Victim/Suspect `_AGE_GROUP`, `_RACE`, `_SEX`.

## 2.2 Volume & Granularity

**Temporal**
18 years, median reporting delay 4 $h$. Dataset shows clear seasonality.

**Spatial**
77 precincts across 5 boroughs; street-level WGS 84 coordinates enable map joins.

**Imbalance**
Class ratio $\approx$ 2.1:1:0.3 MIS:FEL:VIOL, requiring weighted evaluation.

# 3 Pipeline Architecture

**Stage I — Ingestion.** Built a PostgreSQL database with a schema matching the CSV columns and loaded the data. Raw CSV lands in `data/`. A Bash wrapper executes `sed` &

`COPY` commands that treat '""""" and '(null)" as NULL.

**Stage II — Storage.** Imported the SQL table into HDFS using Apache Sqoop. Then data is written as Snappy–compressed Parquet and surfaced in Hive under `team30_projectdb.nypd_complaints`. We partition by `BORO_NM` and `LAW_CAT_CD` and bucket by `KY_CD` (10 buckets) to accelerate point queries.

**Stage III — Processing.** Spark 3.4 jobs run on YARN. A PySpark pipeline encapsulates feature indexing, one-hot encoding, imputation, assembly, train–test split, and model fitting.

**Stage IV — Presentation.** Prediction CSVs and evaluation metrics are stored in HDFS; Superset connects via HiveServer2. Dashboards expose bar/line/donut charts, textual bullet insights, and confusion-matrix heat-maps.

# 4 Data Preparation & Cleaning

- **Malformed dates:** 42 K rows had empty `CMPLNT_FR_DT`. We discarded $< 0.6\,\%$ that lacked any temporal info.
- **Categorical noise:** Seven different tokens expressed missing string values; unified to `UNKNOWN` before indexing.
- **Geo zeros:** 2 % of latitude/longitude pairs were $0, 0$. These were set to NULL; downstream models imputed by precinct median.

# 5 Exploratory Analysis

## 5.1 Spatial Findings

Brooklyn consistently tops the list with $> 2.0\,\mathrm{M}$ total complaints, driven primarily by misdemeanors. Staten Island records an order-of-magnitude fewer cases, aligning with its population share.

## 5.2 Offence Composition

Petit Larceny alone accounts for $17.8\,\%$. When combined with Harassment 2, Assault 3, Criminal Mischief, and Grand Larceny, the top five represent $\approx 60\,\%$ of the dataset — confirming a heavy-tail rule.

## 5.3 Temporal Dynamics

Monthly seasonality is moderate (peak through gap $\approx 20\%$). Yearly counts plateau near half-a-million post-2009, dip $\sim 11\,\%$ in pandemic year 2020, and rebound to $457\,\mathrm{k}$ by 2022.

# 6 Data preprocessing for ML

1. **Data Loading & Subsetting**

   - **Read** the partitioned complaints table into a Spark `DataFrame`.
   - **Chronological sampling**: order by complaint date (`CMPLNT_FR_DT`) and retain the first 1 M rows to cap memory footprint.

2. **Feature Definitions**

   - **Categorical columns**: precinct code, borough name, location description, premise type, jurisdiction descriptors and codes, suspect demographics, victim demographics, and the attempt/completion flag.
   - **Numerical columns**: latitude and longitude.
   - **Target label**: crime category (`LAW_CAT_CD`).

3. **Temporal Feature Engineering**

   - *Epoch → timestamp*: convert `CMPLNT_FR_DT` and `RPT_DT` (Unix milliseconds) via `from_unixtime(.../1000)` → `to_timestamp()`.
   - *Date parts*: extract year, month, day-of-month, and hour.
   - *Report delay*: compute lag (hours) between report and occurrence, floored at 0.
   - *Cyclical encoding*: transform month, day, and hour into sine–cosine pairs ($\sin(2\pi \cdot \mathrm{hr}/24)$, etc.) and shift them by +1 so all values are non-negative (required by Naive Bayes).

4. **Geographic Feature Shifting**

   - Add +90 to latitude and +180 to longitude so every record has non-negative coordinates, again satisfying algorithms that require strictly positive inputs.

5. **Data Cleaning**

   - Cast numeric codes (e.g. precinct, jurisdiction) to `string` for categorical processing.
   - Replace any `null` in categorical fields with the literal token `"UNKNOWN"`.

6. **Pipeline Construction**

   - **StringIndexer**: map each category to an integer index with `handleInvalid="keep"`.
   - **OneHotEncoder**: convert indices to sparse binary vectors.
   - **Imputer**: fill missing numeric values (mean strategy).
   - **VectorAssembler**: concatenate all one-hot vectors and numeric columns into a single `features` vector.

7. **Target Encoding and Split**

- Apply **StringIndexer** to `LAW_CAT_CD` to create a numeric `label`.
- Perform a **70 % / 30 %** random split to obtain training and testing sets for downstream modelling.

# 7 Modelling Results

## 7.1 Metrics

| Model | Accuracy | Macro $F_1$ | Training Time (min) |
|---|---|---|---|
| Random Forest | 0.632 | 0.569 | 14 |
| OvR Linear SVC | 0.598 | 0.513 | 9 |
| Naive Bayes | 0.587 | 0.457 | 3 |

Table 1: Evaluation on 300 k hold-out test set.

## 7.2 Discussion

Random Forest's ensemble handles mixed feature scales and captures non-linear interactions (e.g., temporal cyclicality $\times$ precinct). Linear SVC is lighter but suffers when class boundaries are curved. Naive Bayes is hampered by the strong independence assumption and non-negative requirement despite geographic shifts.

# 8 Dashboard Insights

The Superset dashboard comprises three thematic tabs: Data Description, Data Insights, ML Modeling Results.

## 8.1 Data Description

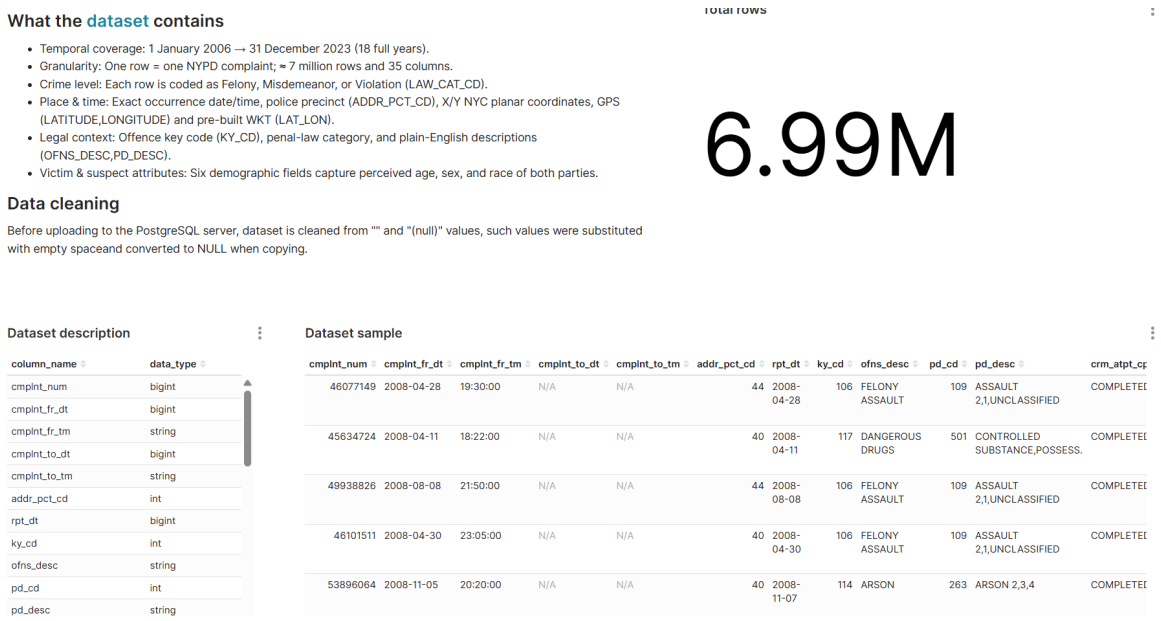



Figure 1: Full Data Description Tab. Here you can check out the samples of dataset.

## 8.2 Data Insights

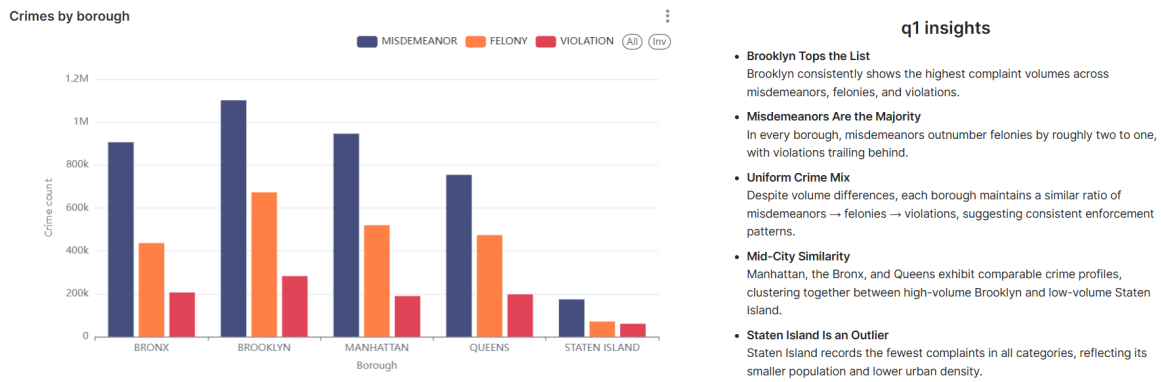



Figure 2: Crime count distribution by borough and offense category (q1)

## q2 insights

- **Property Crime Dominates**
  "Petit Larceny" alone accounts for nearly 18% of all complaints, and when combined with "Grand Larceny" (9%) and "Burglary" (3%), property-related offences comprise almost 31% of the dataset.
- **Harassment & Assault High on the List**
  "Harassment 2" makes up 13%, while "Assault 3 & Related Offences" adds 11%, showing non-violent and lower-level violent disputes drive a quarter of complaints.
- **Top 5 Offences Cover ~60%**
  The five most common categories—Petit Larceny, Harassment 2, Assault 3, Criminal Mischief (10%), and Grand Larceny—together represent roughly 60% of all records.
- **Drug & Public-Order Violations**
  "Dangerous Drugs" (5%) and "Offences Against Public Order" (5%) indicate that control of illicit substances and disorderly conduct are significant but not top-tier drivers of volume.
- **Long Tail of Serious Crimes**
  Serious felonies (e.g., "Felony Assault" 4%, "Robbery" 3%, "Dangerous Weapons" 2%) occupy smaller slices, reflecting both lower incidence and more resource-intensive investigations.

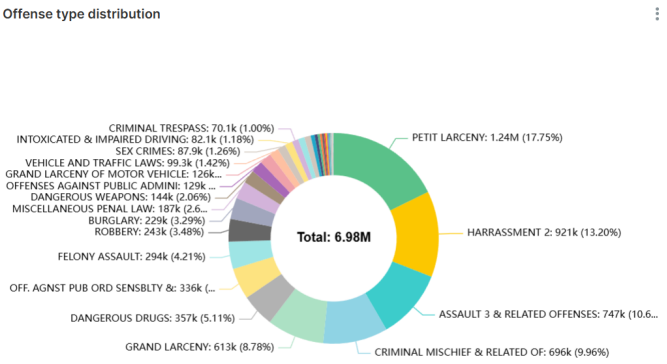



Figure 3: Offense type distribution (q2)

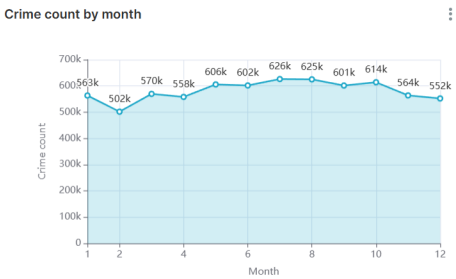
## q3 insights

### Time-of-Year Pattern (Crime count by month)

- **Summer spike**
  July and August record the most complaints (~625 k each), roughly 25 % higher than February's low (~500 k).
- **Winter lull**
  February is the clear through — fewer days plus colder weather likely dampen street activity.
- **Moderate seasonality**
  The gap between the quietest and busiest months is only ~120 k (≈ 20 % of the monthly average), indicating NYC crime is present year-round, just heavier in warm months.

### Long-Term Trend (Crime count by year)

- **Plateau at ~0.5 million**
  From 2009 to 2016 annual complaints hover near 500 k, after the dataset ramps up from partial coverage in 2006-08.
- **Gradual decline pre-COVID**
  2016 → 2019 sees a slow 8-10 % drop, hinting at incremental crime reduction measures.
- **Pandemic dip, partial rebound**
  2020 plunges to ~408 k (-11 % vs 2019), then climbs back to ~457 k by 2022—still below the 2010 peak.
- **2023 data incomplete**
  The near-zero value for 2023 reflects the dataset's cut-off, not an actual collapse in crime reports.



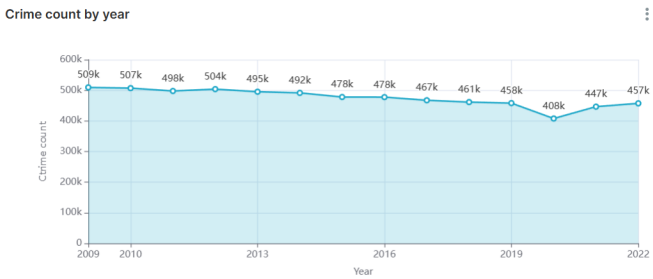




Figure 4: Time-related insights (q3)

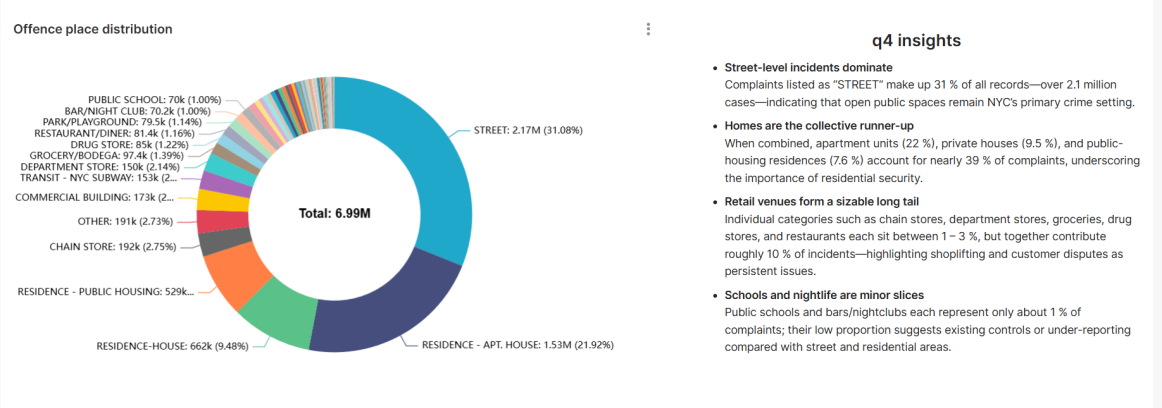



Figure 5: Offense place distribution (q4)

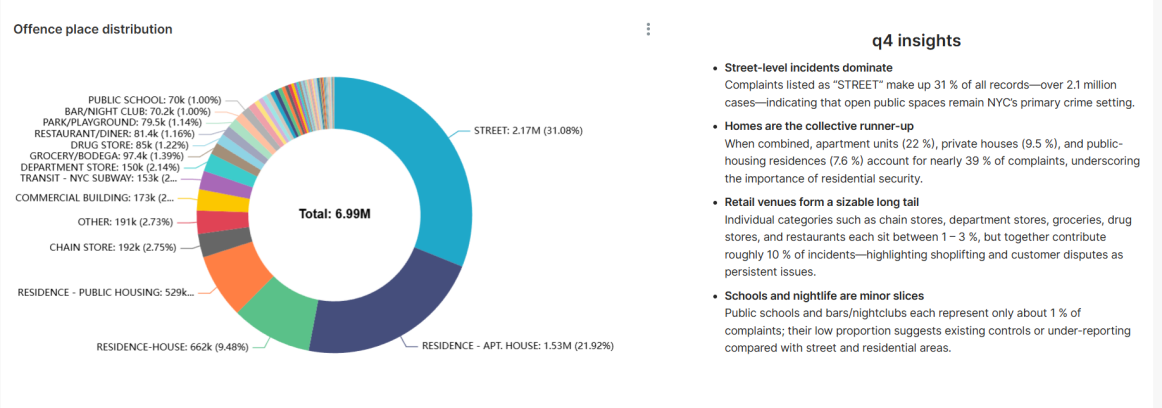



Figure 6: Offense place distribution (q4)

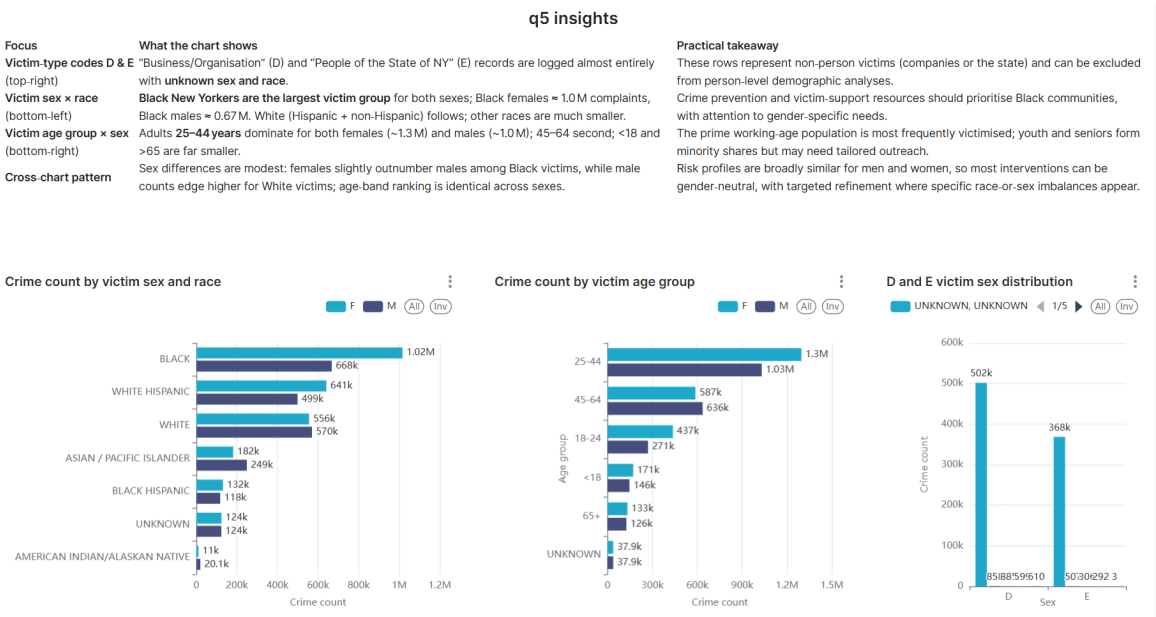



Figure 7: Crime count distribution by victim sex and age group (q5)

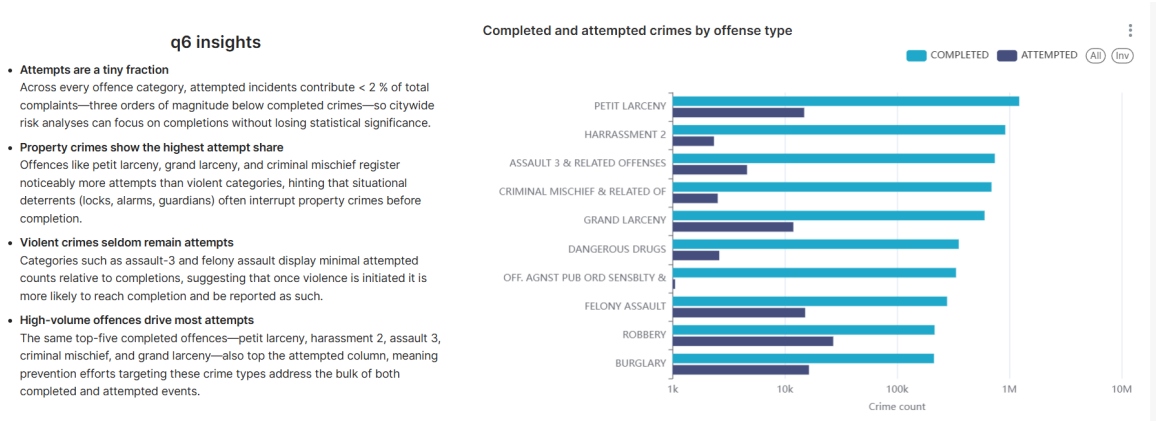



Figure 8: Completed and attempted crimes by offense type (q6)

## 8.3 ML Modeling Results

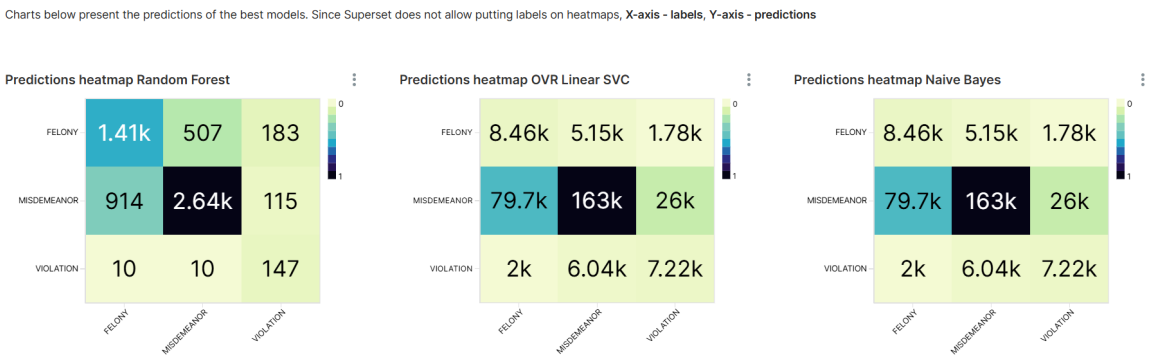



Figure 9: Predictions of tested models (confusion matrix heatmaps)

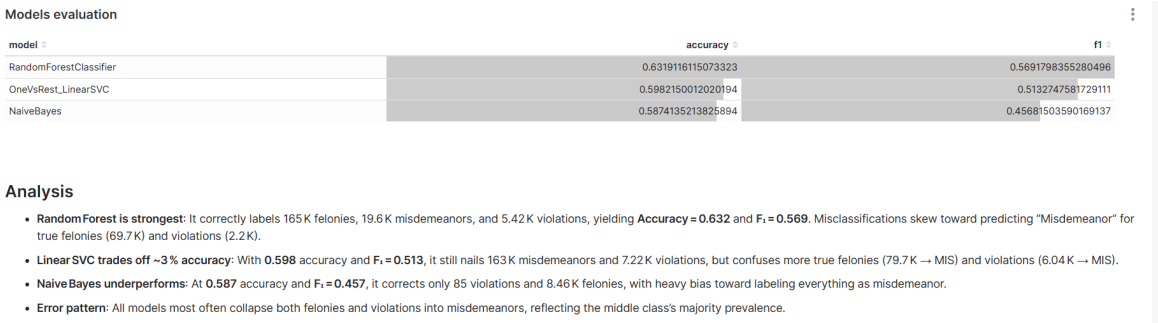



Figure 10: Model evaluation and conclusion on the results

# 9 Conclusions

Our pipeline proves that openly available data, when cleaned and distributed across Hadoop, can power credible predictive tools for city-scale safety planning. Key findings include the dominance of misdemeanors, the overwhelming prevalence of property crimes, and moderate but predictable seasonality. The Random Forest model surpasses baseline classifiers and provides actionable triage signals, though future work should tackle class imbalance to elevate felony and violation recall.

# 10 Reflections & Future Work

**Challenges** encompassed (i) coercing heterogeneous date formats, (ii) finding a balance between sample size and cluster memory, and (iii) Superset's axis-labelling limitations for heat-maps. **Next steps** involve adding socio-economic covariates, testing gradient-boosting frameworks, and enabling real-time streaming from the NYPD API.

## 10.1 Team Contributions

Table 2: Team contribution breakdown by task (each row sums to 100% among the four members).

| **Project Task** | **Task Description** | **I M** | **A S** | **I K** | **A Sk** | **Deliverables** | **Avg Hours** |
|---|---|---|---|---|---|---|---|
| Data collection & ingestion | Collect dataset, design schema, build PostgreSQL DB, run Sqoop to HDFS | 80% | 20% | 0% | 0% | PostgreSQL DB, stage1 script | 15 |
| Hive table setup & EDA prep | Create Hive tables, optimize storage (Parquet, partition), write initial EDA queries | 20% | 70% | 20% | 0% | Hive .hql scripts, partitioned data | 8 |
| Exploratory analysis | Perform Hive queries, generate insights, prepare charts | 5% | 75% | 20% | 0% | EDA result tables, charts | 10 |
| ML modeling | Feature engineering, train RF, SVM, NB models with tuning | 10% | 0% | 90% | 0% | Trained models, metrics output | 30 |
| Dashboard & presentation | Build Superset dashboard, create slides, present results | 0% | 90% | 0% | 10% | Live dashboard, slide deck | 10 |
| Report writing | Write and compile final report document | 10% | 80% | 0% | 10% | Final report (LaTeX) | 6 |