Chicago Crime Data Analysis and Domestic Crime Prediction

1st Yogesh Sivakumar

MSc. Business Analytics

University of Limerick

24150029@studentmail.ul.ie

ABSTRACT

This report is a description of an end-to-end big data analysis, machine learning, and prediction pipeline to determine if a crime is domestic or not. This was done using the Chicago Crime Dataset. The key steps involved in this project were data cleaning, feature engineering, model building, and evaluation. The value of big data and machine learning towards improving public safety becomes evident through these and the prospects that lie ahead for improved feature engineering and dynamic crime forecasting.

1. INTRODUCTION

Urban crime remains a large issue for society, impacting public safety, policy making, and resource use by law enforcement. Applying predictive analytics to criminology is gaining popularity, enabling proactive actions before crime occurs rather than simply responding afterward. In this context, applying historical crime data to forecast certain types of criminal activity, such as domestic violence cases, can assist in concentrating efforts and improving the community.

The objective is to build a precise model to predict whether a crime is domestic or non-domestic based on the key influencing factors in the Chicago crime dataset. To process the high volume of data, the Databricks Community Edition platform was used, and PySpark, SparkSQL, and SparkML were employed in processing, transforming, and modeling data.

2. DATASET AND PREPROCESSING

2.1 Dataset Overview

The data utilized in this project was obtained from the Chicago Data Portal and contains public records of crime incidents during the years ranging from 2000 to 2025. The Chicago Crime dataset was selected as it is sizable, represents real-life scenarios, and contains comprehensive details, thereby presenting a solid foundation for developing and verifying machine learning models. It contains precise information regarding every reported crime, including its primary type, location, timing, outcomes, etc., relative to police interventions.

The data set is approximately 1.8GB in size and has over 8 million instances, making it extremely well-suited for large-scale, real-world machine learning tasks. Considering the complexity and size of the data set, PySpark was used for processing the data, and SparkSQL was used for effective querying, aggregation, and feature exploration. These approaches helped in quickly transforming and altering the huge data set without relying on local computer resources.

The primary objective of this project is to predict whether an offense reported was domestic or non-domestic based on provided features.

2.2 Column Description

The columns present in the dataset, their data types, and their description are summarized in the table below:

Column Name	Data Type	Description
ID	Integer	Unique identifier for each reported crime
Case Number	String	Unique case reference number
Date	String	Date and time when the crime occurred
Block	String	Approximate address where the crime occurred
IUCR	String	Illinois Uniform Crime Reporting Code
Primary Type	String	Primary classification of the crime (e.g., THEFT, BATTERY)
Description	String	Detailed description of the crime
Location Description	String	Describes the type of location where the crime occurred (e.g., STREET, RESIDENCE)
Arrest	Boolean	Indicates if an arrest was made during the incident
Domestic	Boolean	Indicates whether the incident was domestic-related
Beat	Integer	Smallest police geographical patrol area
District	Integer	Larger police administrative area
Ward	Integer	City council district number
Community Area	Integer	Official community area number
FBI Code	String	Standardized FBI classification for the crime
X Coordinate	Integer	X-coordinate (spatial mapping)
Y Coordinate	Integer	Y-coordinate (spatial mapping)
Year	Integer	Year when the crime occurred
Updated On	Timestamp	Date of last record update
Latitude	Double	Geographical latitude of the crime location
Longitude	Double	Geographical longitude of the crime location

Location	String	Combined field of latitude and longitude

2.3 Data Cleaning

A lot of time was spent on data cleaning and preparation for model training. The raw data had problems like missing values, duplicates, inconsistency, and redundant records. All of these problems were resolved by using PySpark DataFrame operations and SparkSQL queries.

Initially, the records with default or missing values in significant fields (Case Number, Primary Type, and ID) were excluded to maintain dataset integrity. Primary fields such as Primary Type, Date, Latitude, and Longitude were mandatory, so records with missing values for these fields were removed. Secondary missing fields like Location Description was filled with "UNKNOWN", while missing Ward and Community Area values were filled with -1 to indicate unknown entries without losing important data.

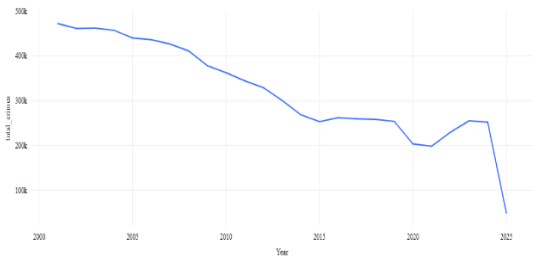
We removed duplicate events using the Case Number to avoid duplication bias from duplicate records. We enforced location consistency by only taking events that took place within latitude ranges of 41.0 to 42.0 and longitude ranges of -88.0 to -87.0, which are the official geographical boundaries of Chicago.

The categorical text fields (Primary Type, Location Description) were normalized to uppercase for consistency during feature engineering. The Date field was originally a string, converted to a Timestamp to be able to extract additional time-related features such as the hour of the incident, day of the week, and month. Summary statistics were performed to gain an initial understanding of the dataset's structure and distributions

2.4 Data Visualisations

To get an overview and support feature understanding, several visualizations were created based on the cleaned dataset. These visualizations show the trends over time, crime distributions, and key categorical breakdowns across the city of Chicago. The visualizations and their interpretations are summarized below:

- **Crime Trends Over Time (Line Graph)**:

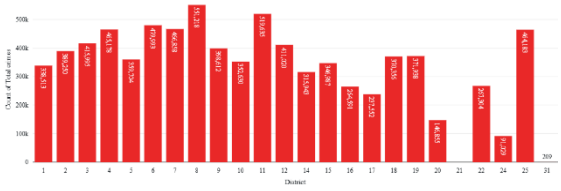


A line graph plotting total crime counts per year revealed a consistent decline in reported crimes.

- **Crime Type Distribution (Word Cloud)**:

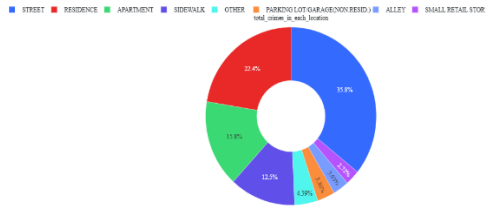
A word cloud visualization showed that 'Theft', 'Battery', and 'Criminal Damage' are the most frequent crime types, followed by 'Narcotics', 'Assault', 'Motor Vehicle Theft', and 'Burglary'. This highlights that a few crime categories dominate the overall crime landscape in Chicago.

- **Crime by Police District (Bar Chart)**:



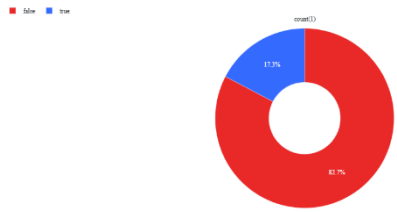
A bar chart analysis indicated that Districts 8 and 11 recorded the highest number of total crimes, each exceeding 500,000 incidents. Districts 7 and 25 also had significantly high crime counts.

- **Location of Crimes (Donut Chart)**:



Analysis of crime locations showed that the majority of crimes occurred on streets (35.8%), followed by residences (22.4%), apartments (15.8%), and sidewalks (12.5%). Public spaces and residential areas emerged as the primary hotspots for criminal incidents.

- **Domestic vs Non-Domestic Crimes (Donut Chart)**:



The final visualization revealed that 82.7% of crimes were non-domestic, while 17.3% were classified as domestic-related. Although domestic crimes represent a minority, they still constitute a significant share requiring focused attention.

These visualizations collectively provided valuable initial insights regarding the distribution, severity, and enforcement patterns associated with crime incidents in Chicago. They also informed the selection and prioritization of features for the predictive modeling phase.

2.5 Feature Engineering

Once the data cleaning and EDA were done, feature engineering tasks were performed to construct additional attributes that could enhance the model's predictive power. Three time-related features were extracted from the timestamped Date column: the hour indicating the time of day when the crime occurred, the day of the week, and the month of the incident. These time variables were created to identify possible trends and to verify whether these factors influence domestic crimes.

Categorical columns were transformed to prepare them for those machine learning algorithms that require numbers. Features like Primary Type, Location Description, Community Area, and Ward were converted to numbers using PySpark's StringIndexer. Similarly, the target variable Domestic was converted to a string type and then converted to a binary label suitable for a classification algorithm, with 1 indicating domestic crimes and 0 indicating non-domestic crimes. VectorAssembler from PySpark was then used to combine all the chosen features into one feature vector. This created a structured dataset that was ready for training predictive models.

To ensure that the engineered features were still relevant and not redundant, statistical tests were conducted prior to building models.

Correlation analysis was conducted among the numeric variables (Hour, DayOfWeek, and Month) to check whether multicollinearity is an issue. The correlation coefficients produced were close to zero, indicating very weak linear relationships among these variables, and thus justify their use in the model simultaneously without redundancy.

Correlation Matrix			
Feature	**Hour**	**Day of Week**	**Month**
Hour	1.00000000e+00	1.55647099e-02	-3.48408119e-04
Day of Week	1.55647099e-02	1.00000000e+00	-2.57195026e-03
Month	-3.48408119e-04	-2.57195026e-03	1.00000000e+00

To find out why the categorical predictors are important, we performed a Chi-square test between each of the categorical variables and the target variable. The Chi-square test is used to tell us whether distributions of categorical variables are different from what we would have expected if there were no associations between the variables.

The Chi-square test results:

Feature	Degrees of Freedom	Chi-Square Statistic	p-value	Interpretation Target variable = Domestic (labelled as 0 and 1)
Primary Type	35	2,375,594.08	0.0	Strong dependency on the target variable
Location Description	217	1,483,082.78	0.0	Strong dependency on the target variable
Community Area	78	190,255.61	0.0	Strong dependency on the target variable
Ward	50	177,718.91	0.0	Strong dependency on the target variable

The null hypothesis of independence for all of the features was rejected in all the tested features because their p-values in the Chi-Square tests were 0.0. This means that these categorical features have significant differences with the target variable and therefore can be used in the final modeling process.

Statistical significance testing utilized the standard testing framework, where an upper limit of 0.05 for p-values was set. Since all p-values discovered were much less than this value, the features being tested were considered to be significantly related to the home crime label.

3. MODEL IMPLEMENTATION AND RESULTS

3.1 Machine Learning Model Selection

The target variable, Domestic, had a moderate class imbalance, with just about 17% of incidents being domestic offenses. A Random Forest Classifier was chosen to address this. Random Forests are appropriate for dealing with imbalanced data since they generate numerous decision trees using samples from the original data, then vote on the result, which works to balance errors across classes.

3.2 Data Splitting, Model Training, and Optimization

The processed data was divided randomly in the ratio 80:20 into training and testing datasets by Pyspark's random split function. Optimization techniques, caching and partitioning, were utilized to process the large dataset (~1.8GB, over 8 million rows) effectively. Caching was done on the DataFrame after processing before model training to keep in memory interim results and avoid duplicated computation in cyclical tasks. Repartitioning was used to distribute the data across multiple partitions, thereby increasing parallelism and speeding up the transformation and training processes.

Model training was carried out through PySpark's machine learning libraries. Random Forest Classifier was trained with the merged feature vector as input and the indexed label column (Domestic) as output. Starting configurations, i.e., number of trees and maximum depth, were chosen based on best practices for large real-world datasets.

3.3 Model Evaluation

A comprehensive set of evaluation metrics was employed, including:

- **Accuracy**: 0.8876 ~ **88%**
- **Precision**:0.8794083391284314 ~ **87.9%**
- **Recall**: 0.8875762499003269 ~ **88%**
- **F1-Score**: 0.8791941739277723 ~ **87.9%**

These results indicate that the model achieved strong and balanced performance across all key evaluation metrics, demonstrating its effectiveness in predicting domestic crime incidents.

3.4 Findings from Model Evaluation

The Random Forest Classifier achieved approximately 88% accuracy on the test set, which suggests that it is highly capable of differentiating between domestic and non-domestic crime incidents. Precision of 87.9% implies that when the model predicts a crime to be domestic, it is right nearly nine times in ten. The 88% Recall shows that the model correctly detected the great majority of actual domestic crime cases, keeping false negatives to a minimum. The F1-score of roughly 87.9% reflects a high level of balance between Precision and Recall, confirming to the model's ability in dealing with the moderately imbalanced dataset.

Overall, these results show that the selected machine learning method, combined with appropriate data preparation, feature engineering, and optimization, produced a very effective predictive model that can be applied to real-world applications.

4. OPTIMIZATION AND TUNING

Performed optimization tasks such as cache() and repartition(). Cache was implemented on the cleaned dataset, which was indexed, because further processing of the data to ML model implementation requires repeated access to the data, so caching helped reduce computation time. Similarly, repartition was implemented before training the dataset, so that the data is split into multiple partitions, reducing the computational workload while implementing the random forest classifier algorithm.

5. CONCLUSION

This project aimed to predict whether a crime in Chicago that was reported was because of domestic issues or not, using machine learning algorithms on a big real-life dataset. Performed high-level data preparation, including cleaning, feature creation, and feature selection, to get the data ready for modeling. Because our target variable was fairly unbalanced, we chose a Random Forest Classifier since it can deal with these issues and works well with this type of data. Caching and repartitioning techniques were employed to speed up processing while training the models.

There were a number of challenges encountered throughout the project. Collaboration on Databricks Community Edition was difficult due to the limited computing resources and the fact that it used to slow down while processing a dataset of over 8 million records. We planned to perform Cross-validation and hyperparameter tuning, and we tried, but the Databricks community edition didn’t support it because the Spark cluster gets detached automatically after a period of one hour.

Future improvements might include moving to a stronger computing platform, trying out advanced group methods like Gradient Boosting, and using data balancing methods such as SMOTE to improve the detection of smaller groups. Additionally, using MLflow in Databricks to keep track of experiments, monitor model versions, and automate tuning of model settings could greatly enhance experiment management and repeatability.

The study illustrated the way in which big data computer programs and machine learning algorithms can be applied in predicting crime. This offers insightful information about trends in public safety.