CIND 820 Big Data Analytics Project

Crime Prediction Using Classification Approaches

Student Name: Mariela Marmanillo Mendoza

Student Number: 501188993

Supervisor: Tamer Abdou

Date of Submission: April 1, 2024

Ryerson University

Table of Contents

List of Tables and Figures

Abstract

Crime in Toronto has been relatively low compared to other larger cities in North America; however, there is a sensation that incidents are increasing more and more in Toronto, leading to an unsafe feeling among the population. Every day and every week, the news shows incidents of robberies, break-and-enters, and assaults in public places and private places where, before, no one ever knew about that type of event. Even when, in most cases, only material losses are experienced (for example, a car theft or theft of artifacts and jewelry in break-ins), fear and insecurity are growing among the population.

The intention of this study is to analyze crime occurrences in Toronto from 2014 to 2023, which are included in the database provided by the Toronto Police Service, where incidents have been reported since 2014 by time, date, and location – the nearest road intersection node to protect the privacy of parties involved in the occurrence. The Major Crime Indicators (MCI) in the database include Assault, Break and Enter, Auto Theft, Robbery, and Theft Over, excluding sexual violations, homicides, and shootings. According to the description of the database, the data collected has been determined through a police investigation as founded, meaning that the offence did occur or was attempted to occur. The database has 372,899 observations and was last updated on January 11, 2024 (Toronto Police Service, 2024).

In this project, the research questions are the following:

1. What is the comparative assessment of accuracy among various machine learning classification algorithms, including decision tree, K-nearest Neighbour, and Naïve Bayes models, when analyzing crime incidents in Toronto?

2. Which information of a crime occurrence allows for more accurate predictions: the temporal patterns—time of the day, day of the week, or month-- or the spatial patterns--location, premises, or neighborhood?
3. How the results obtained from this research contribute to understand historical crime patterns and develop a plan with better-targeted interventions to improve community safety.

By answering these questions, we could help decision-makers understand historical crime patterns and develop a plan with better-targeted interventions to improve community safety.

I propose developing a predictive model that accurately classifies MCIs based on historical crime occurrence data in the dataset described above. Supervised machine learning method, such as the decision tree algorithm for classification tasks, is a suitable tool in this database labeled with categorical MCI features. Its performance will be compared with Naïve Bayes and K-nearest Neighbours models. In this case, the models will be trained and tested on a dataset that contains the desired categorization (The Chang School of Continuing Education, Class Notes, 2023).

Literature Review, Data Description, and Approach

Literature Review

Several studies have focused on the analysis of crime trends as it has become a significant issue affecting almost all countries around the world. All research in this area pretend to provide useful insights for city planners and law enforcement agencies in developing effective crime prevention plans.

Crime prediction can be reached using different methods, including statistical, visualization, unsupervised learning, and supervised learning methods. Machine learning algorithms are widely used in crime prediction (Saeed and Abdulmohsin, 2023). According to Safat et al. (2021), crime prediction involves assessing the accuracy of past reported crimes, while forecasting involves predicting future crime trends.

Numerous research projects have been conducted to predict crime types, crime rates, and crime hotspots using datasets from different areas in major cities around the world (Safat et al., 2021). Machine learning techniques have proven effective in forecasting spatial crime data (Saeed and Abdulmohsin, 2023). For this project, various research studies that explore the relationship between crime occurrences and different machine learning techniques were reviewed. One study was conducted for Toronto utilizing the same database as this project, while the remaining studies centered on cities within the United States, with one exception in India.

Using the Major Crime Indicators (MCI) database provided by the Toronto Police Service, Esfahani and Esfahani (2023) studied crime rate trends in Toronto from 2014 to 2022. They considered the temporal scales such as year, month, week, day and hour to predict the number of crime incidents per month, and forecast them for 2023 and 2024 using a deep learning method (neural network model). This method was compared with Naïve Bayes and weighted moving average models. In order to evaluate the models' performance and accuracy in predicting crime occurrences, three metrics--Mean Square Error (MSE), Mean Absolute Error (MAE), and Mean Absolute Percentage Error (MAPE)—were used. The results showed that deep learning model outperformed the other two, concluding that crime events have increased in Toronto from 2014 onwards, with this trend expected to continue throughout 2024.

The Naïve model is a simple method that forecast time series by assuming that the future observations will be the same than the last value and does not contemplate further changes of features in the data or futures adjustments, while the weighted moving average model considers relevant observations and assign weights to them based on certain characteristics to improve forecasting accuracy (Esfahani and Esfahani, 2023).

Almanie et al. (2015) analyzed crime datasets for the cities of Denver, Colorado and Los Angeles, California, to predict crime types on specific dates (month, date and time) and at a particular location in a data-mining model. The study combined the Denver dataset with the neighborhood demographics dataset existed for that city. Data transformation, discretization, as well as an apriori algorithm were applied in order to identify recurring patterns and improve model accuracy, while Naïve Bayesian and decision tree classification methods were used for crime-type prediction. These supervised learning algorithms were built using Scikit-Learn library tool, applying a 5-fold cross validation strategy on both models to compare accuracy. The Naïve Bayes model shown the best performance in crime prediction, with accuracies of 51% and 54% for Denver and Los Angeles respectively, while the decision tree classifier showed accuracies of only 42% and 43%.

Vaquero Barnadas (2016) used k-nearest neighbours (KNN), Parzen windows, and artificial neural networks on the San Francisco, California crime dataset to determine which of them works best solving the category classification problem. The dataset contains 39 different crime types or categories. Due to the huge amount of data, the k-means algorithm was applied for clustering and reducing the size of the database. The study focused on classification and prediction, with the neural network algorithm proving to be the most accurate.

Safat et al. (2021) compared different machine learning methods for crime prediction using the large crime datasets for Los Angeles, California and Chicago, Illinois. The methods include logistic regression, support vector machine (SVM), Naïve Bayes, k-nearest neighbors (KNN), Decision Tree, multilayer perceptrom (MLP), Random Forest and XBoost algorithms for prediction accuracy. Also, a deep learning model named long-short term memory (LSTM) was applied for time-series analysis, and the autoregressive integrated moving average (ARIMA) model for forecasting crime rate and crime density areas. The LSTM evaluation was reported using the performance metrics Root Mean Squared Error (RMSE) and MAE. The datasets contained 35 and 39 types of crime, with the results showing that XGBoost outperformed other models with accuracies of 94% and 88% in Chicago and Los Angeles, respectively, followed by KNN with 88% and 89%.

Hussain and Aljuboori (2022) explored crime report dataset for Boston, Massachusetts to predict incident crime types using decision tree, logistic regression and Naïve Bayes classification models. The results showed that the decision tree classifier performed better than the other machine learning techniques, concluding that crime location alone is enough to build an accurate model, as crime amount and type are strongly related to location.

According to Saeed and Abdulmohsin (2023), supervised learning approaches are predominantly used in crime prediction studies, with logistic regression being the most robust method, showing an accuracy of 90% against lower percentages of decision tree, Naïve Bayes and random forest models. In their study, logistic regression algorithm was used to predict the correlation between burglar crimes and various factors, including time of day, day of week, barriers, connectors, and repeated victimization. However, this model was ineffective when applied to large geographical areas.

Kumar et al. (2020) applied the k-nearest neighbours (KNN) model to predict crime in the city of Indore, Madhya Pradesh, India. The research goal was to predict the type of crime likely to occur in a particular area based on the date, time and location. Mean Absolute Error (MAE) and Root Mean Square Error (RMSE) were used as reference metrics to compare the results with those of a previous study that determined a different optimal value for k.

The examined studies aim is focused on comparative study between supervised learning algorithms using classification techniques of machine learning to predict the crime type in different cities with unique features. All researchers' intention is to use the information for criminal investigations and crime prevention strategies. Table 1 summarizes the classifying method used by those research projects and the method that showed the best performance. For this project, we are using the three more commonly used classification algorithms: Decision Tree, Naïve Bayes and KNN.

Table 1. Summary of classification techniques used by previous studies

Study	City	Methods	Best Method
Almanie et al. (2015)	Denver, CO & Los Angeles, CA (USA)	Naïve Bayesian and Decision Tree	Naïve Bayes
Vaquero Barnadas (2016)	San Francisco, CA (USA)	K-nearest Neighbours (KNN), Parzen Windows, and Artificial Neural Networks	Neural Network
Kumar et al. (2020)	Indore, MP (India)	K-nearest Neighbours (KNN)	KNN
Safat et al. (2021)	Los Angeles, CA & Chicago, IL (USA)	Logistic Regression, Support Vector Machine, Naïve Bayes, K-nearest Neighbors (KNN), Decision Tree, Multilayer Perceptrom, Random Forest and XBoost Algorithms	KNN followed by XBoost
Hussain and Aljuboori (2022)	Boston, MA (USA)	Decision Tree, Logistic Regression and Naïve Bayes	Decision Tree
Esfahani and Esfahani (2023)	Toronto, ON (Canada)	Neural Network (Deep Learning), Naïve Bayes and Weighted Moving Average	Neural Network

All the studies reviewed address cases similar to the one that will be carried out in this project, including attributes related to the temporality of the crime such as time, day, month, year, and the location of the crime scene, which includes latitude and longitude. However, the categories or labels assigned for the types of crimes are not the same as the categories in our dataset, with the exception of the Esfahani and Esfahani (2023) study, which uses the same Major Crime Indicators (MCI) database as in this project. This discrepancy arises because the police department of each city assigns its categories depending on the characteristics of the impact zone. Furthermore, Esfahani and Esfahani (2023) combine deep learning methods with machine learning methods, whereas in this project, only use machine learning classification algorithms are suggested to be used. Therefore, it can be stated that we are not replicating any study exactly.

Data Description and Exploratory Data Analysis (EDA)

The dataset used in this study includes all Major Crime Indicators (MCIs) retrieved from the Toronto Police Service public safety data portal. It is published in accordance with the Municipal Freedom of Information and Protection of Privacy Act, which means that the Toronto Police Service has taken care of the privacy of every person involved in the reported crime occurrences; hence, no personal information is provided nor the exact location of the crime scene is revealed for any reason (Toronto Police Service, 2024). Furthermore, before downloading the database, full knowledge of the Open Government License for Ontario was taken, which literally indicates that the user is free to "Copy, modify, publish, translate, adapt, distribute or otherwise use the Information in any medium, mode or format for any legal purpose" (Ontario, 2013).

The dataset is available here: https://data.torontopolice.on.ca/datasets/TorontoPS::major-crime-indicators-open-data/about. It is updated on a regular basis. For this study, it was downloaded on

January 22, 2024, reflecting the last upload to the database as of January 11, 2024. The selected dataset contains 372,899 observations since 2014. According to the description, the collected data has been determined through a police investigation as founded, which means that the offence did occur or was attempted to occur.

The MCIs include Assault, Break and Enter, Auto Theft, Robbery, and Theft Over, while excluding sexual violations, homicides, and shootings. Appendix 1 shows the indicators glossary. The information is collected at the crime and/or victim level; therefore, several observations may have the same identification number EVENT_UNIQUE_ID associated with the different MCIs used to categorize the incident (Toronto Police Service, 2024). Appendix 2 shows the dictionary of attributes.

The exploratory data analysis (EDA) report for the selected dataset was prepared using the ydata-profiling library in Python. The following link https://github.com/marielamarmanillo/CIND820 allows access to the repository for this project where all the files including Python code are stored. Figure 1 shows an overview of the dataset statistics, noteworthy indicating a small number of missing values, representing less than 0.1% of the total values, which suggests the dataset's quality. Additionally, there are 31 attributes in this dataset, categorized into four types: numeric, text, date-time, and categorical.

Figure 2 shows key considerations regarding the studied data. First, the feature OBJECTID is highlighted for its unique values and uniform distribution. This is expected, as the OBJECTID attribute serves as a simple enumeration of dataset observations, starting from 1. This variable is omitted from the Toronto Police Service column description, so it will be dropped during data cleaning and preprocessing.

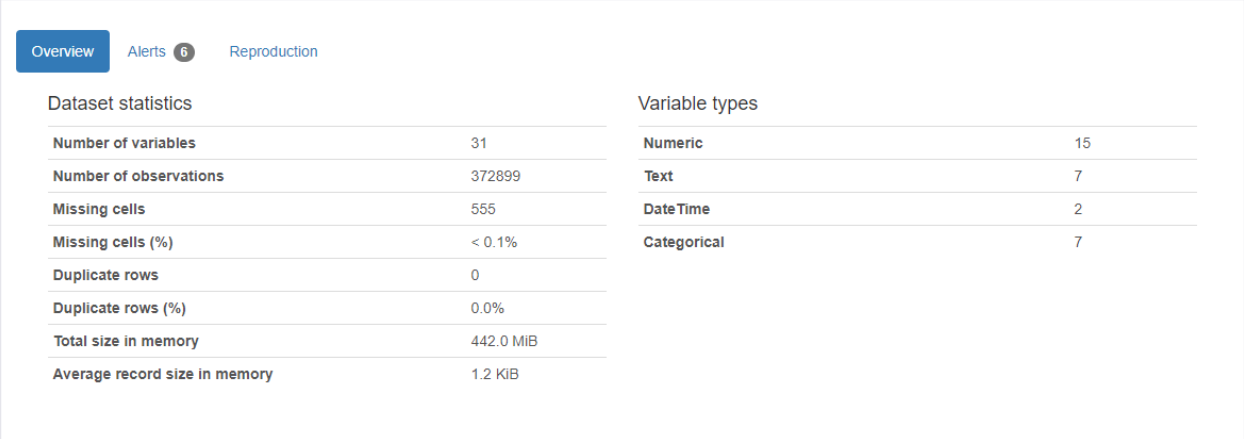


Figure 1. Dataset Overview from *ydata-profiling* Report

The REPORT_HOUR and OCC_HOUR variables include some percentage of zeros due to the reporting protocol of the Toronto Police Service (Figure 2). In their 24-hour time format, crimes reported or occurred within the first hour of the day are denoted as 0, corresponding to the time range from 12:00 am to 12:59 am.

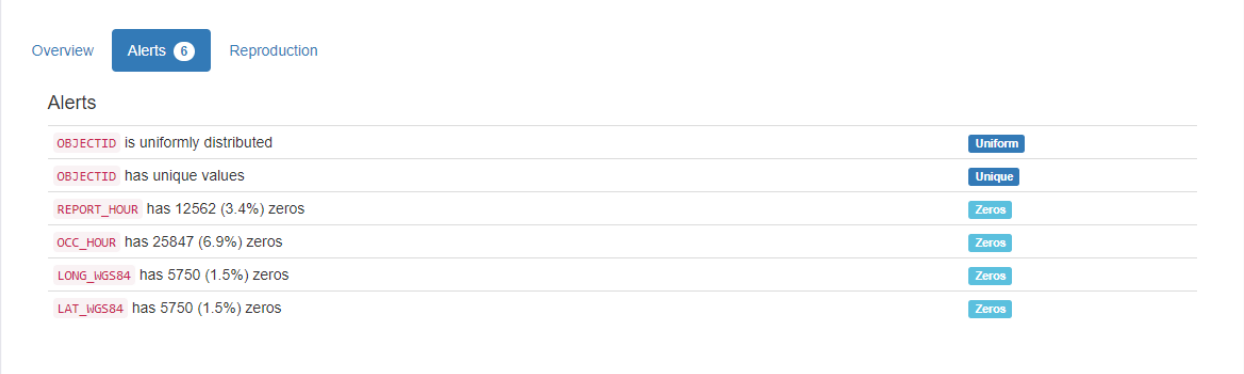


Figure 2. Dataset Alerts from ydata-profiling Report

The longitude (LONG_WGS84) and latitude (LAT_WGS84) variables, which both together indicate the coordinates of the nearest intersection where the crime occurred, show a small percentage of zero values (Figure 2). Specifically, there are 5,750 zero values in each column. As these instances involve both longitude and latitude being assigned zero at the same time, these

rows will be dropped. Given the relatively small number of observations, omitting these rows does not significantly impact the overall analysis of the dataset.

Figure 3 exhibits the overview of the categorical variable of interest, Major Crime Indicators. It has 5 categories or labels as described earlier in this section. No missing values were found. The events categorized as Assault constituting more than half of the total crime occurrences (53.1%). Following Assault, Break and Enter is the next significant category, representing 18.8% of the total. Auto Theft and Robbery are the next two notable categories, making up 15.7% and 9.1% of the total. Finally, Theft Over is the least frequent category, accounting for only 3.3%.

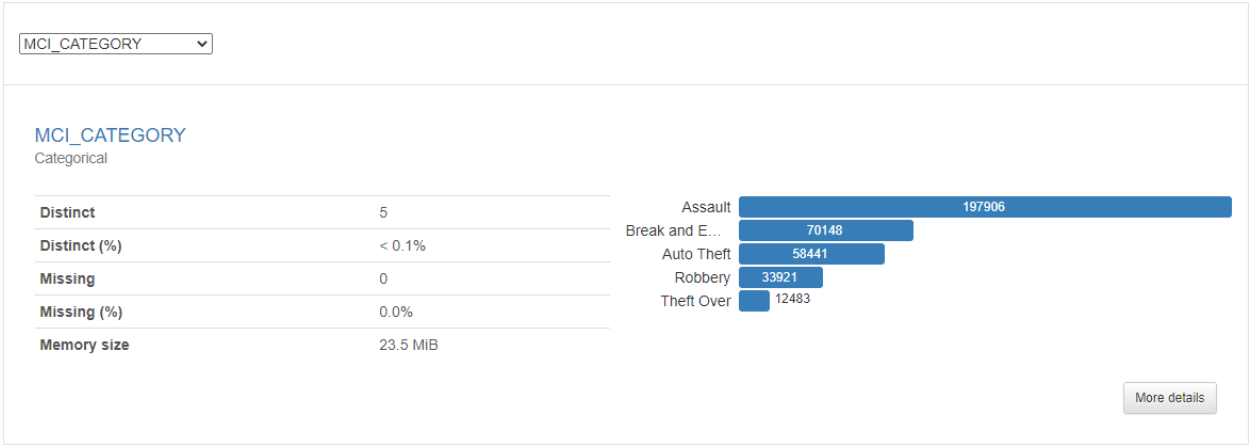


Figure 3. MCI Categorical Variable Overview from ydata-profiling Report

The variable containing the year in which the crime occurs ranges from 2000 to 2023. However, based on the histogram shown in Figure 4, 99.5% of the data falls within the years 2014 to 2023, while only 0.4% falls between 2000 and 2013. Additionally, there is no data for 111 records, representing 0.1% of the total data in this attribute. Therefore, records for years with insignificant data will be omitted from the analysis.

Figure 4. Year of Crime Occurrence Overview from ydata-profiling Report

Figure 5 exhibits the summary of the selected dataset.

```
DataFrame info:
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 372899 entries, 0 to 372898
Data columns (total 31 columns):
 #   Column              Non-Null Count   Dtype
---  ------              --------------   -----
 0   X                   372899 non-null  float64
 1   Y                   372899 non-null  float64
 2   OBJECTID            372899 non-null  int64
 3   EVENT_UNIQUE_ID     372899 non-null  object
 4   REPORT_DATE         372899 non-null  object
 5   OCC_DATE            372899 non-null  object
 6   REPORT_YEAR         372899 non-null  int64
 7   REPORT_MONTH        372899 non-null  object
 8   REPORT_DAY          372899 non-null  int64
 9   REPORT_DOY          372899 non-null  int64
 10  REPORT_DOW          372899 non-null  object
 11  REPORT_HOUR         372899 non-null  int64
 12  OCC_YEAR            372788 non-null  float64
 13  OCC_MONTH           372788 non-null  object
 14  OCC_DAY             372788 non-null  float64
 15  OCC_DOY             372788 non-null  float64
 16  OCC_DOW             372788 non-null  object
 17  OCC_HOUR            372899 non-null  int64
 18  DIVISION            372899 non-null  object
 19  LOCATION_TYPE       372899 non-null  object
 20  PREMISES_TYPE       372899 non-null  object
 21  UCR_CODE            372899 non-null  int64
 22  UCR_EXT             372899 non-null  int64
 23  OFFENCE             372899 non-null  object
 24  MCI_CATEGORY        372899 non-null  object
 25  HOOD_158            372899 non-null  object
 26  NEIGHBOURHOOD_158   372899 non-null  object
 27  HOOD_140            372899 non-null  object
 28  NEIGHBOURHOOD_140   372899 non-null  object
 29  LONG_WGS84          372899 non-null  float64
 30  LAT_WGS84           372899 non-null  float64
dtypes: float64(7), int64(8), object(16)
```

Figure 5. Dataset Structure

Project Approach

Figure 6 represents the workflow and overall methodology of this project.

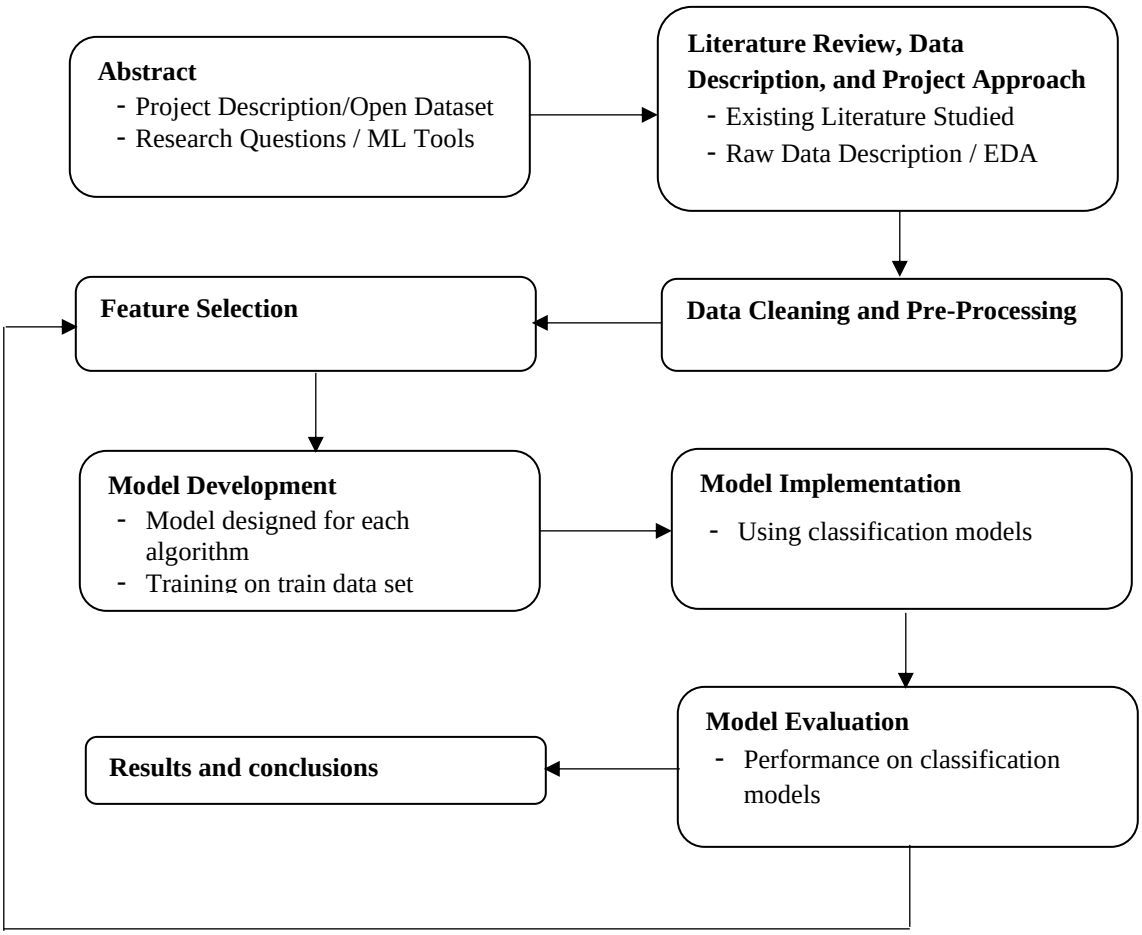


Figure 6. Project Methodology Workflow

Data Cleaning and Pre-Processing

Cleaning the selected dataset involved eliminating duplicate rows, dropping some variables, omitting missing values, and filtering the observations that were recorded on years other than the period 2014 to 2023. Thus, the study only focused on analyzing data from January 1, 2014 to December 31, 2023, spanning 10 years of crime data.

- Removing Duplicate Rows

  According to the dataset description provided by the Toronto Police Services, some crime events were reported by different individuals involved in the incident, such as the victim, the victimizer, and witnesses. As a result, there were multiple reports made by different people about the same event. To address this, only one observation of each event was retained. A total of 47,920 duplicate rows were identified and removed based on the offence identifier (EVENT_UNIQUE_ID). After removing these duplicates, the dataset was left with 324,979 rows and 31 columns Following the removal of duplicates, the dataset now comprises 324,979 rows and 31 columns, down from the original 372,899 rows × 31 columns.

- Omitting Missing Values

  After removing duplicate rows, 88 missing values were identified in each of the columns corresponding to the year, month, day, day of the year, and day of the week of occurrence. Interestingly, the same row is the one with data missing for each of the above-mentioned columns. Given that only 88 rows out of the 324,979 total contain missing values, these rows were omitted at this stage; therefore, the new data subset has 324,891 rows and 31 columns.

- Dropping Rows with Crimes that Occurred in years other than the period 2014-2023

  Since this study covers the analysis of crime events from January 1, 2014 to December 31, 2023, the crime data observations with years of occurrence from 2000 to 2013 were removed from the dataset. In this case, 1,171 rows were eliminated and the dataset now shows 323,720 rows and 31 columns.

- Dropping rows that contains NSA in variables NEIGHBORHOOD

  In the columns representing the new neighborhood structures, a subset of observations contained missing values denoted by "NSA," totaling 5,066 entries. Given their relatively small proportion compared to the overall dataset, these entries were removed. Consequently, the dataset now consists of 318,654 rows and 31 columns.

After the described arrangements, all relevant attributes to be used in this study do not include missing values. Furthermore, object variables such as month of crime occurrence, day of the week, premises type, and MCI category were converted to category data type. Similarly, year, day of occurrence, and day of the year underwent conversion from float to integer data type. The new data types are shown in Figure 7.

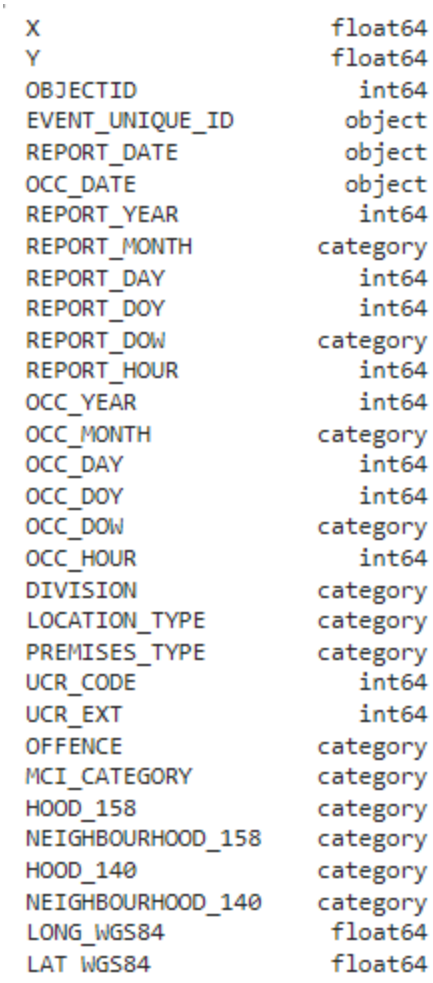
Preliminar Analysis

After cleaning the data, this section presents an analysis of crime trends and patterns. It aims to identify key information such as the most common types of crimes committed in Toronto, trends in crime rates over time, frequently occurring crimes, and crime hotspots based on location, among other factors.

In Toronto, the average number of crime incidents in the last 10 years was 31,865 per year, 2,656 per month, and 89 per day. Table 2 shows the count of different types of crimes (Assault, Auto Theft, Break and Enter, Robbery, and Theft Over) for each year from 2014 to 2023, along with the total sum of all crimes for each year. It is noteworthy that overall, crime rates in Toronto have shown a sustained upward trend, except for the years during the pandemic (2019 to 2021), where

there was a significant decrease. Subsequently, there was a resumed increasing and sharp trend in crime rates, as shown by the trend line in Figure 8.

However, this has not been a trend observed in all types of crimes in this study. Figure 9 shows the trend described above, which is very pronounced in assaults and break-ins; however, car thefts have increased significantly even during the pandemic. These three cases deserve a lot of attention as they are having the most impact. Regarding robbery, cases decreased during the pandemic, but post-pandemic they have not increased to pre-pandemic levels. Finally, theft over cases have remained constant.



X	float64
Y	float64
OBJECTID	int64
EVENT_UNIQUE_ID	object
REPORT_DATE	object
OCC_DATE	object
REPORT_YEAR	int64
REPORT_MONTH	category
REPORT_DAY	int64
REPORT_DOY	int64
REPORT_DOW	category
REPORT_HOUR	int64
OCC_YEAR	int64
OCC_MONTH	category
OCC_DAY	int64
OCC_DOY	int64
OCC_DOW	category
OCC_HOUR	int64
DIVISION	category
LOCATION_TYPE	category
PREMISES_TYPE	category
UCR_CODE	int64
UCR_EXT	int64
OFFENCE	category
MCI_CATEGORY	category
HOOD_158	category
NEIGHBOURHOOD_158	category
HOOD_140	category
NEIGHBOURHOOD_140	category
LONG_WGS84	float64
LAT_WGS84	float64

Figure 7. Variables Data Type

Table 2. Crime Counts by Category and Year (2014-2023)

MCI_CATEGORY	Assault	Auto Theft	Break and Enter	Robbery	Theft Over	Total
OCC_YEAR						
2014	13826	3409	7073	2816	985	28109
2015	14753	3050	6790	2711	1005	28309
2016	15530	2994	6230	2825	989	28568
2017	15907	3280	6733	3031	1134	30085
2018	16418	4223	7460	2867	1224	32192
2019	17210	4712	8298	2680	1297	34197
2020	15169	5061	6750	2061	1152	30193
2021	15821	5844	5477	1687	1018	29847
2022	17708	8602	5810	2058	1342	35520
2023	19797	10781	7206	2307	1543	41634

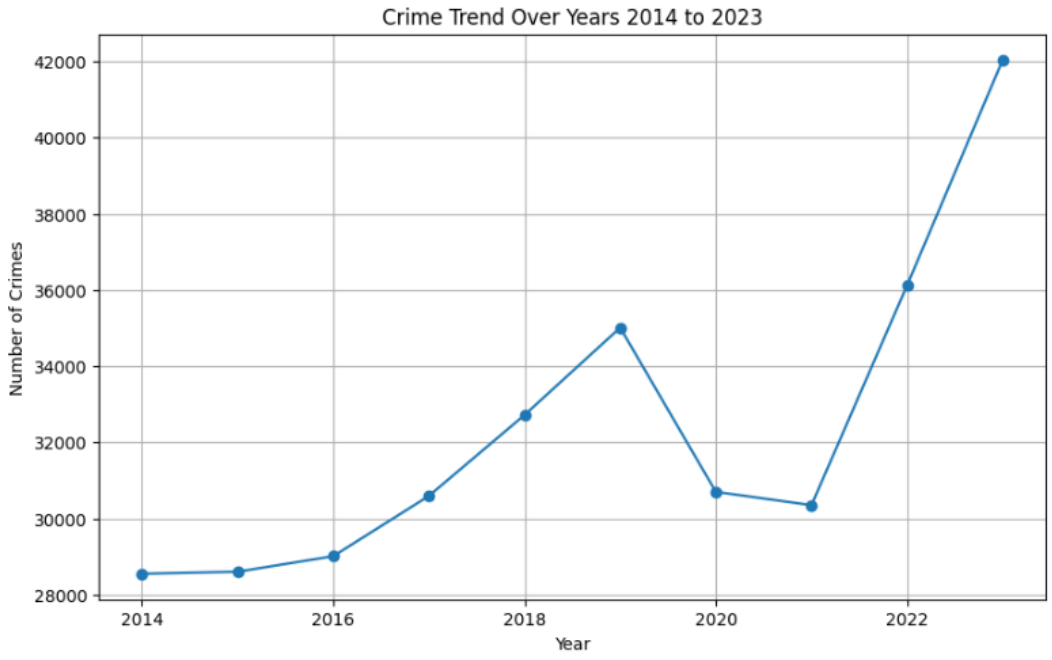


Figure 8. Crime Trend in Toronto (2014 to 2023)

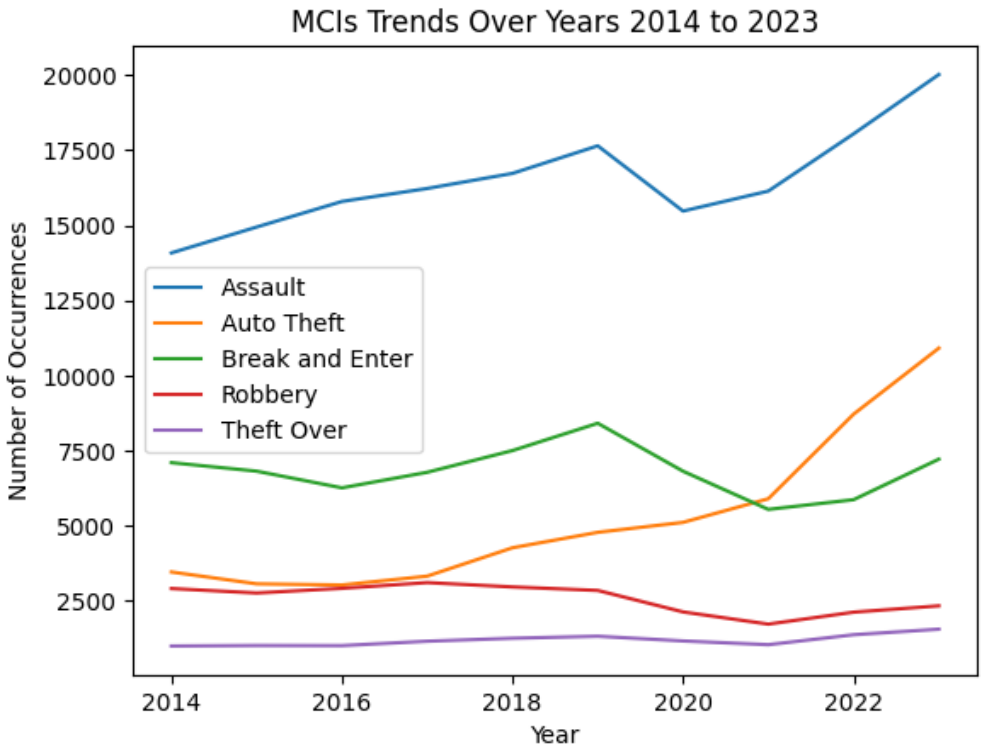


Figure 9. Crime Trends per Major Crime Indicators (2014 to 2023)

Crime occurs most frequently in late winter, late spring, and early summer, and it begins again during the holiday season corresponding to Halloween, Christmas, and New Year's Eve (Figure 10). The first few days of any month are considered to have the greatest number of incidents, and the last few days are considered to have the fewest. It should be noted that the observed peak occurs on the first day of each month (Figure 11). Observing by day, the crime peak is observed at midnight. It is then low during the day, and another peak occurs at noon. Subsequently, it begins to gradually increase from 4 p.m. onwards until night (Figure 12). Additionally, more crimes occur on Sundays than on any other day of the week (Figure 13).

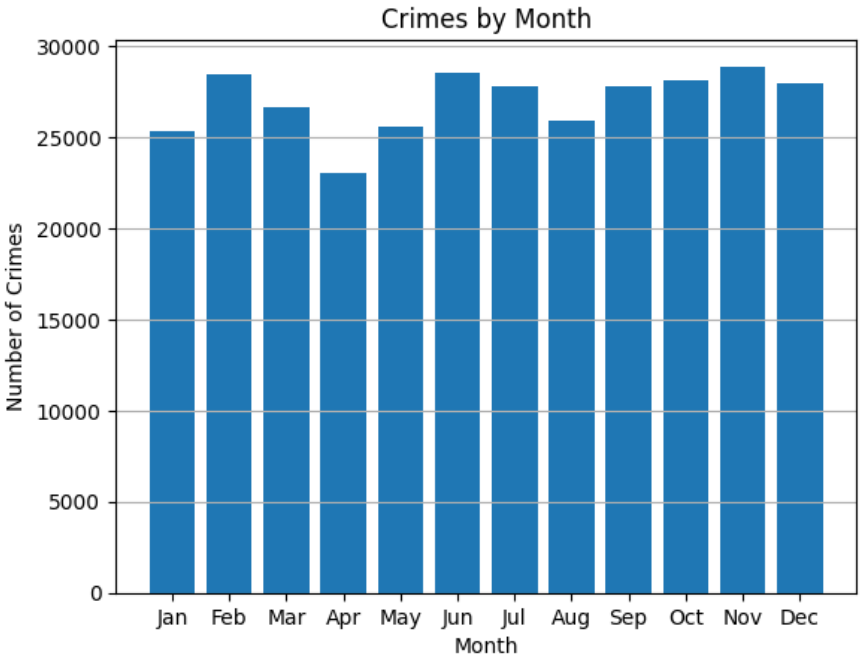


Figure 10. Crimes by Month

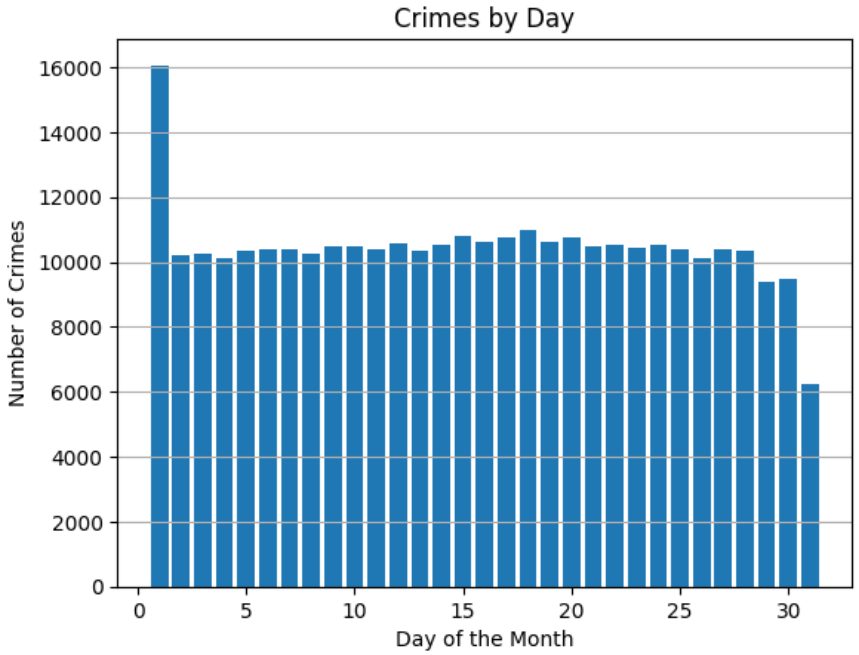


Figure 11. Crimes by Day

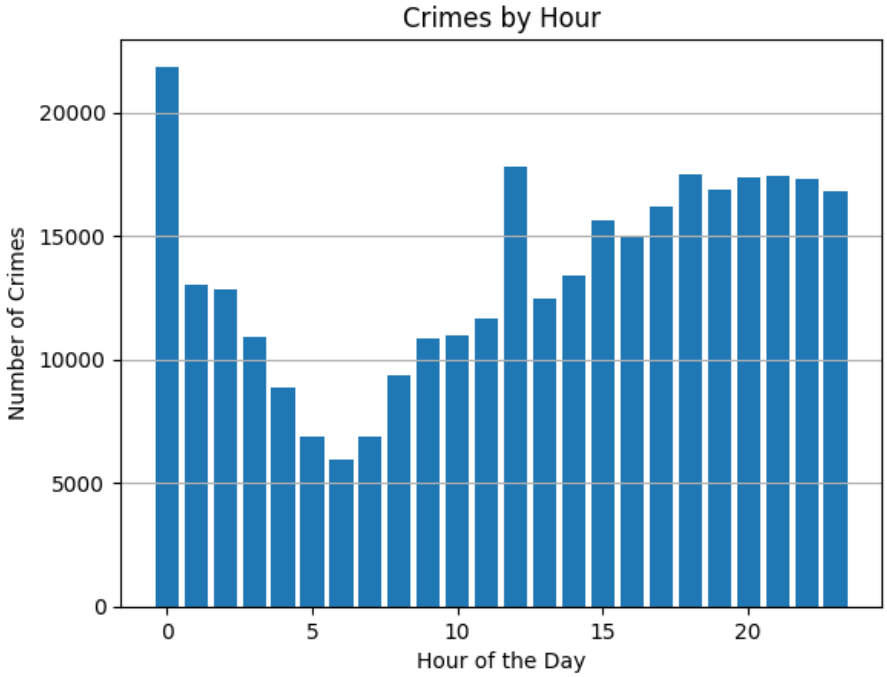


Figure 12. Crimes by Hour

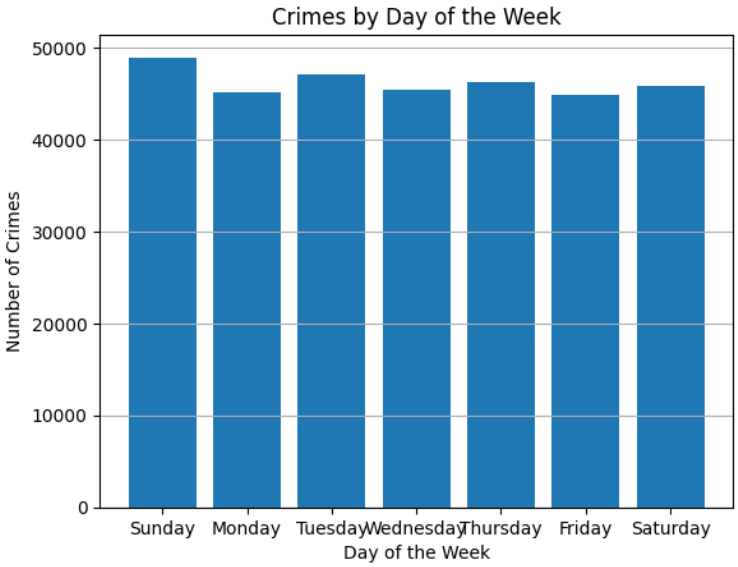


Figure 13. Crimes by Day of the Week

Over the past 10 years, most crimes occurred outdoors, followed by apartments, commercial establishments and homes. To a lesser extent, they occur in transportation facilities and educational centers.

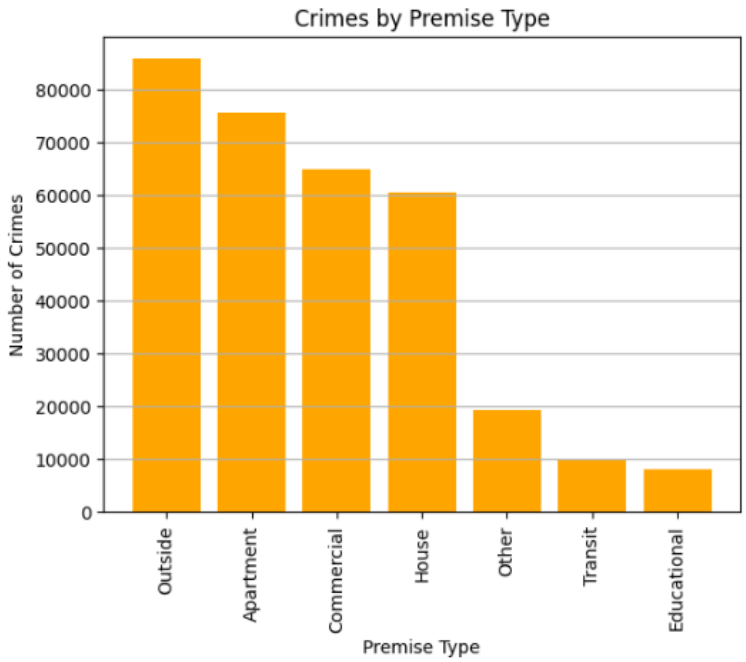


Figure 14. Crimes by Type of Premises

According to Toronto's new neighborhood structure, the most dangerous neighborhood is West Humber-Clairville, followed by Moss Park and Downtown Yonge East, listed as the other two most dangerous (Figure 15). Figure 16 highlights that West Humber-Clairville, the neighborhood with the highest crime rate in Toronto, exhibits a notably elevated incidence of auto theft incidents, surpassing even the prevalence of assault—a category typically predominant across neighborhoods.

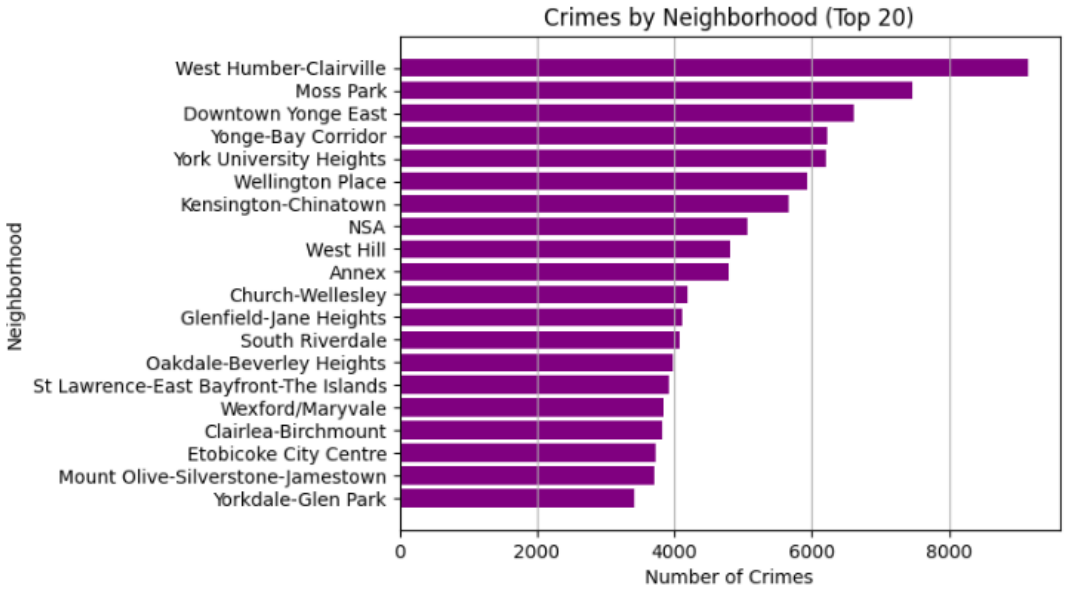


Figure 15. Most Dangerous Neighborhoods

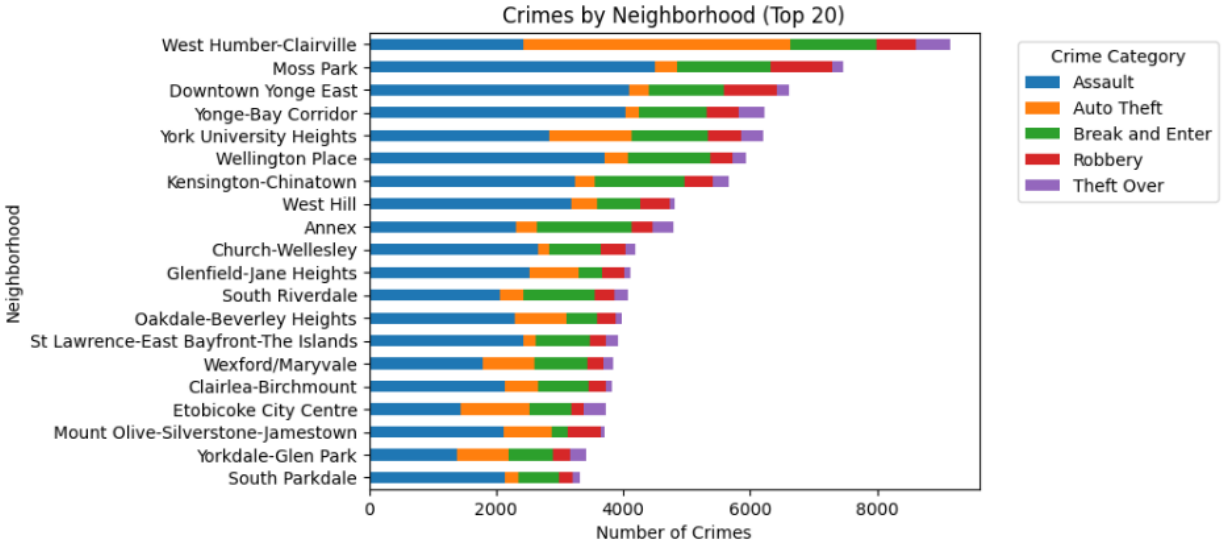


Figure 16. Crime by Category in Most Dangerous Neighborhoods

Among the least dangerous neighborhoods are Lambton Baby Point, Woodbine-Lumsden and Guildwood (Figure 17). The most prevalent crime is assault, as in all of Toronto, followed by car theft and break and enters both in equal magnitude (Figure 18).

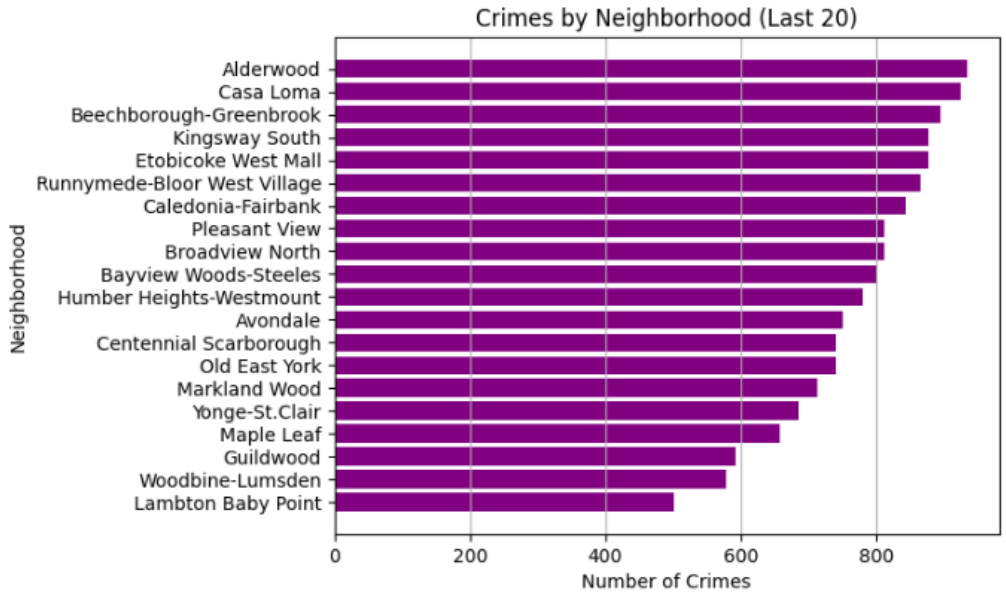


Figure 17. Least Dangerous Neighborhood

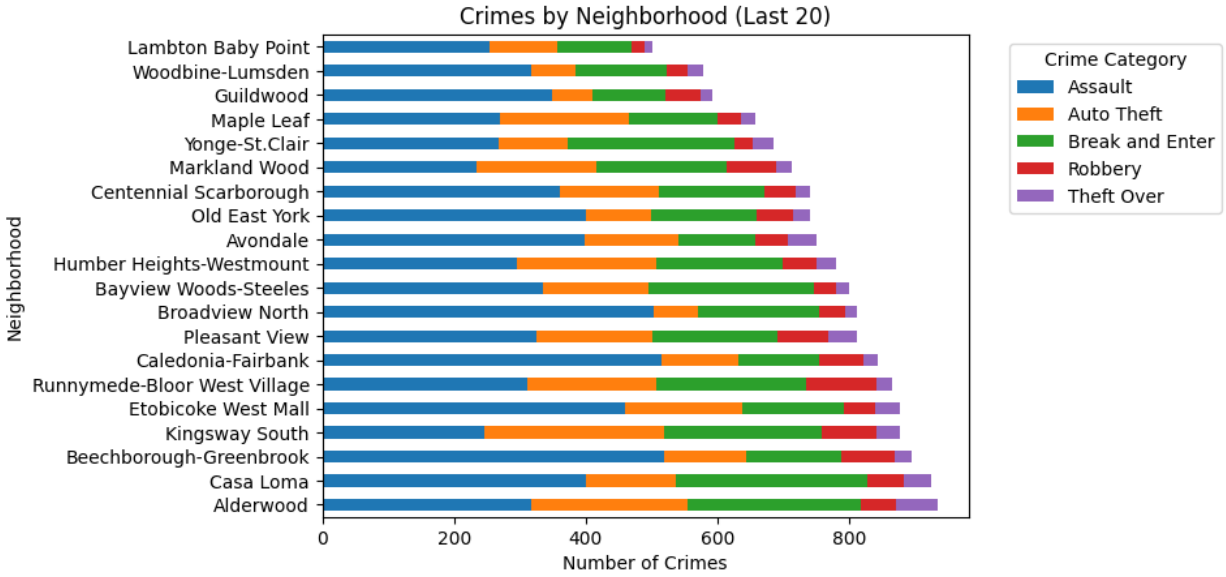


Figure 18. Crime by Category in Least Dangerous Neighborhoods

Figure 19 offers valuable insight into the spatial distribution of crimes throughout Toronto. While the map is based on a subset of the data (the first 1000 rows) for illustrative purposes, the overall dataset yields comparable results. Notably, the visualization highlights a concentration of classified crimes in the downtown area.

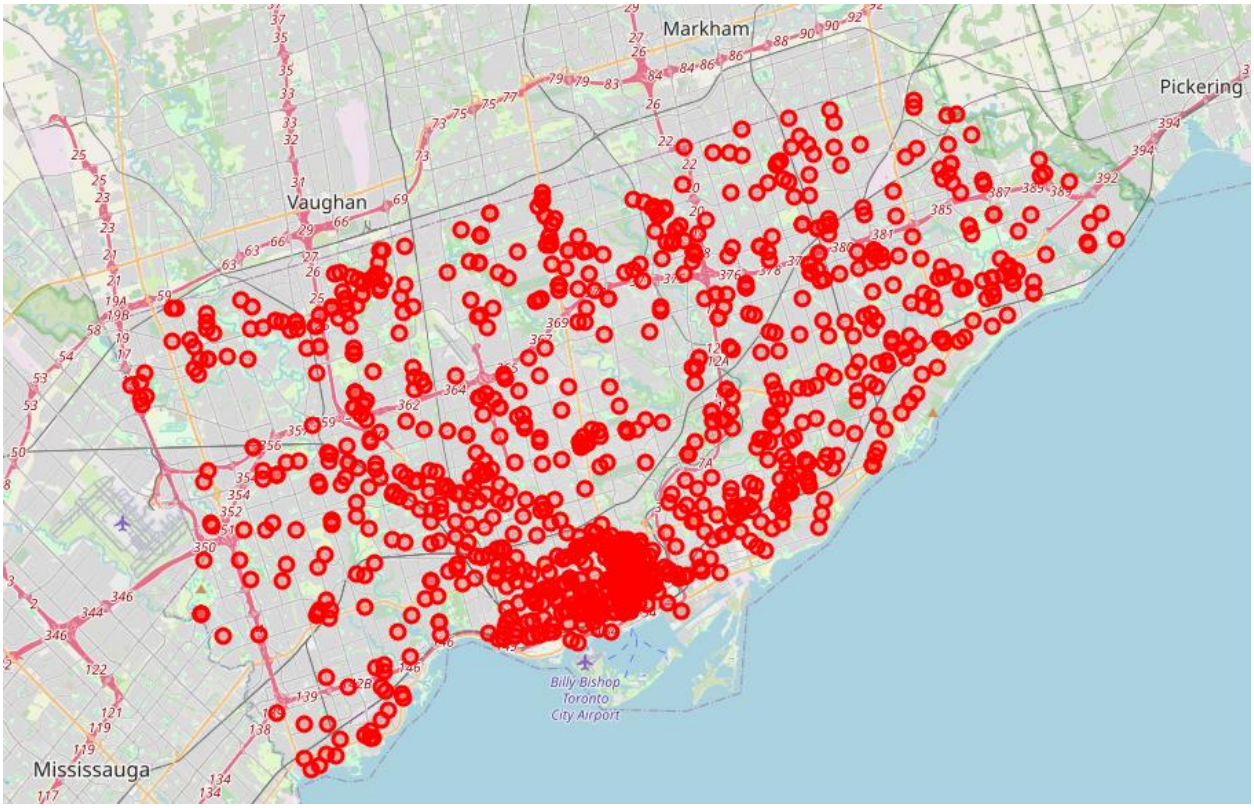


Figure 19. Location of Major Crimes Throughout Toronto

Feature Selection

The goal of feature selection is a data preprocessing technique for selecting the best subset of variables prior to building a machine learning model. It helps to remove irrelevant variables to optimize model construction (Gupta, 2023).

There are a wide variety of feature selection techniques, including filter-based, wrapper-based, and hybrid/embedded families. In this project, specific filter-based feature selection techniques, such as information gain, are employed. Filter-based techniques evaluate features independently of the model that will be built and are less prone to overfitting. This characteristic fits the needs of this study, as three different classification models will be used. Moreover, these techniques are faster and less computationally expensive, although they do not necessarily capture correlations or interactions among features (Gupta, 2023).

Common variants of filter-based feature selection techniques include information gain (capturing the information gain of each variable with respect to the target variable), chi-square test (evaluating the best chi-square scores), and Fisher's score (assessing the best Fisher's scores), among others. In this project, the information gain feature selection method is utilized. This method was chosen due to its effectiveness in ranking variables according to their information gain with respect to the outcome, and because it is usually applied to classification problems, aligning with the project's goals.

Before applying the chosen feature selection technique, a preliminary check is conducted to identify categorical variables in the dataset. Using a threshold of 5%, the ratio of unique values to total values in each column was assessed. This threshold indicates that columns with fewer than 5% unique values may have a limited number of distinct categories, suggesting they are likely categorical. As shown in Table 3, it is determined that most variables met this criterion, except for location-related variables (longitude and latitude), and the unique identifiers (OBJECTID and EVENT_UNIQUE_ID). Given their minimal contribution to the dataset's information and lack of relevance to feature selection, it is decided to drop the ID-related variables at this stage.

Table 3. Categorical Variable Identification

	Variable	Unique Values	Categorical
0	X	19035	False
1	Y	19034	False
2	OBJECTID	318654	False
3	EVENT_UNIQUE_ID	318654	False
4	REPORT_DATE	3652	True
5	OCC_DATE	3652	True
6	REPORT_YEAR	10	True
7	REPORT_MONTH	12	True
8	REPORT_DAY	31	True
9	REPORT_DOY	366	True
10	REPORT_DOW	7	True
11	REPORT_HOUR	24	True
12	OCC_YEAR	10	True
13	OCC_MONTH	12	True
14	OCC_DAY	31	True
15	OCC_DOY	366	True
16	OCC_DOW	7	True
17	OCC_HOUR	24	True
18	DIVISION	17	True
19	LOCATION_TYPE	54	True
20	PREMISES_TYPE	7	True
21	UCR_CODE	22	True
22	UCR_EXT	16	True
23	OFFENCE	50	True
24	MCI_CATEGORY	5	True
25	HOOD_158	158	True
26	NEIGHBOURHOOD_158	158	True
27	HOOD_140	141	True
28	NEIGHBOURHOOD_140	141	True
29	LONG_WGS84	19036	False
30	LAT_WGS84	19036	False

Then, the Label Encoder is used to encode the selected categorical variables. Subsequently, the dataset is split into a 70% training set and a 30% test set before applying feature selection techniques, which are applied only on the training set. This approach helps prevent data leakage and overfitting, ensures accurate model evaluation, and improves computational efficiency. According to Table 4, the features with higher information gain values are the uniform crime report (UCR) code for offense, offense, UCR extension for offense, location type, premises type, latitude, longitude, neighborhood identifier, and neighborhood name. These location-related features are

more informative or influential for predicting the types of major crime indicators. However, the time-related variables such as day, day of the week, day of the year, and month of the crime occurrence provide very low information.

Table 4. Feature Selection - Information Gain Scores

UCR_CODE	1.299937
OFFENCE	1.298079
UCR_EXT	0.941124
LOCATION_TYPE	0.271239
PREMISES_TYPE	0.196592
LAT_WGS84	0.169373
LONG_WGS84	0.168052
Y	0.167878
X	0.167542
HOOD_158	0.061009
NEIGHBOURHOOD_158	0.058703
HOOD_140	0.057055
NEIGHBOURHOOD_140	0.055395
REPORT_HOUR	0.053510
DIVISION	0.035597
OCC_HOUR	0.030758
REPORT_DATE	0.024812
OCC_DATE	0.023041
REPORT_YEAR	0.018328
OCC_YEAR	0.017798
OCC_DOW	0.004942
REPORT_DOW	0.004798
REPORT_DOY	0.003426
REPORT_MONTH	0.003314
OCC_MONTH	0.003166
OCC_DAY	0.002782
OCC_DOY	0.001829
REPORT_DAY	0.001107

Based on the filter-based feature selection results, the eight features with the lower information gain scores are dropped, which are OCC_DOW, REPORT_DOW, REPORT_DOY, REPORT_MONTH, OCC_MONTH, OCC_DAY, OCC_DOY and REPORT_DAY. Additionally, all other variables related to reporting times (REPORT_YEAR, REPORT_DATE, and REPORT_HOUR), as well as the DIVISION where the crime was reported, were also

removed. This decision was made because the main focus of this project is on the times, dates, and locations of the crime occurrence rather than where and when the incidents were reported.

The crime occurrence date variable (OCC_DATE) is also omitted from the analysis because it embraces information contained in other columns corresponding to occurrence year, occurrence month, etc. Although the column related to location type (LOCATION_TYPE) is the fourth with the highest score, it will be also removed as the variable PREMISE_TYPE provides more informative and concise details about the type of crime location (7 categories only) compared to LOCATION_TYPE, which offers overly broad information about the crime scene with 54 categories.

The feature OFFENCE is also omitted from the dataset because it provides information similar to the target variable, but in greater detail. This is evidenced by its 50 categories compared to the 5 categories of the target variable.

Based on what is shown in Figure 20, the variables LAT_WGS84 and LONG_WGS84 will also be dropped. This decision arises from the fact that these variables are essentially identical to variables X and Y.

Furthermore, the old identifiers and names of neighborhood structures in Toronto (HOOD_140, NEIGHBOURHOOD_140) will also be dropped. This decision is based on the fact that the information from the new structure represented in columns HOOD_158 and NEIGHBOURHOOD_158 is nearly identical to the old one, with only slight adjustments (Figure 21).

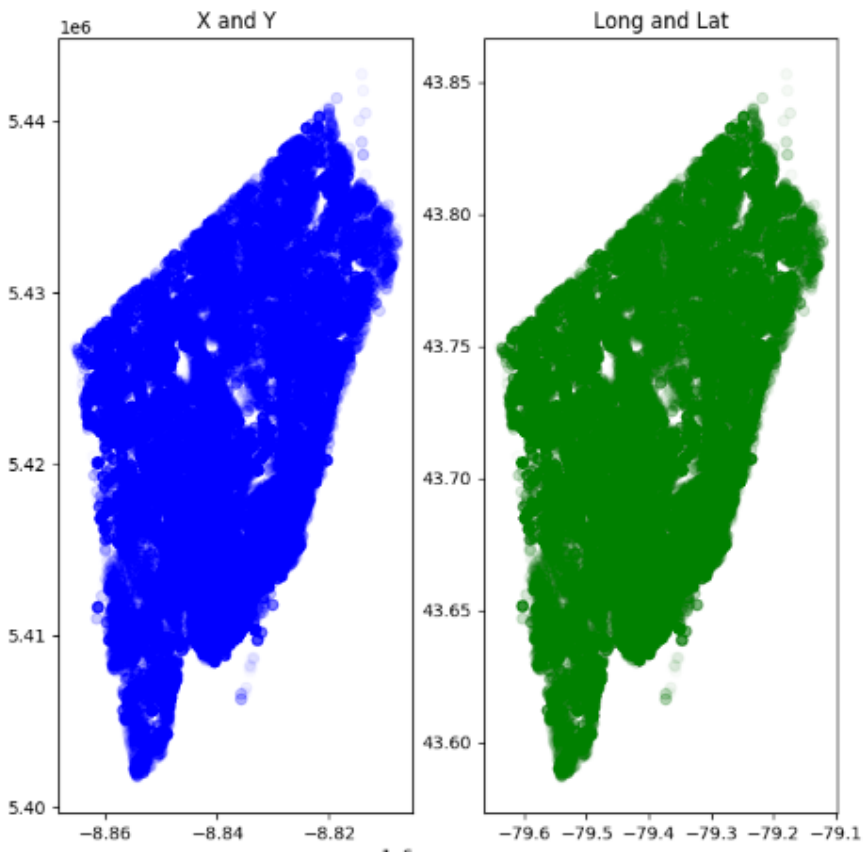


Figure 20. Longitude and Latitude vs X and Y

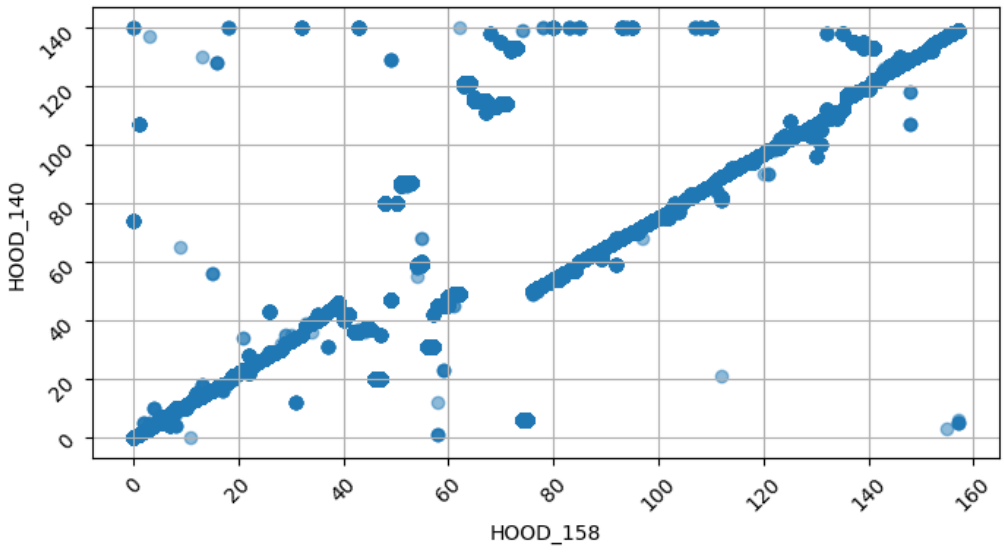


Figure 21. HOOD_158 vs HOOD_140

Finally, Figure 22 illustrates that UCR_EXT and UCR_CODE are two variables highly correlated. This correlation arises because the Uniform Crime Reporting (UCR) code assigns numerical codes to specific types of criminal offenses to standardize crime categories across the country, while the UCR Extension provides additional details or subcategories related to a specific UCR code. Consequently, the UCR extension will be removed from the dataset as it essentially duplicates the information provided by the UCR code but with more detail.

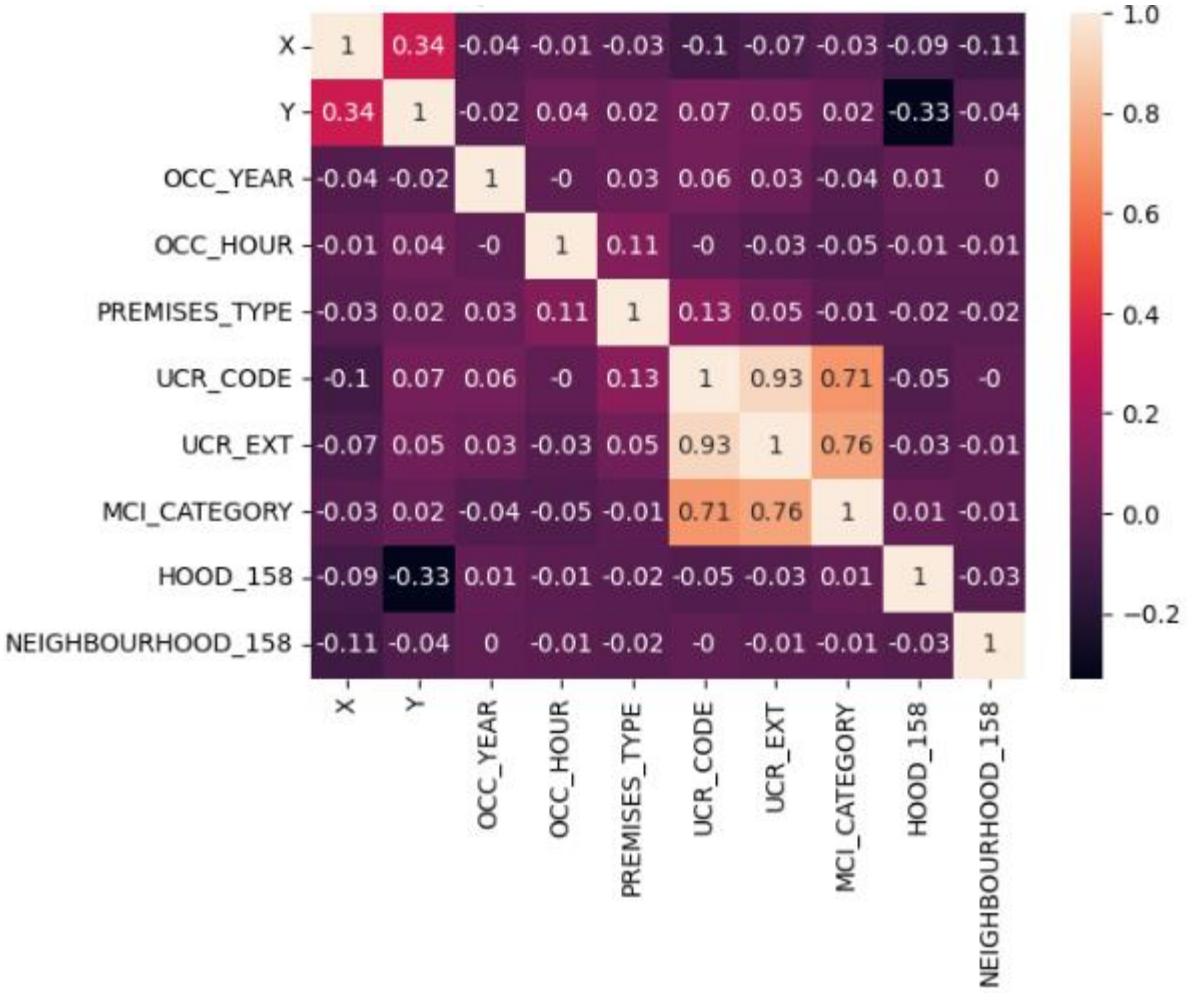


Figure 22. Correlation Heatmap of Variables

After applying filter-based feature selection and dropping the corresponding variables, the dataset size is 318,654 rows × 9 columns.

Model Implementation

Prior to model building, the class imbalance issue should be addressed in the training set. Table 5 shows that the training dataset has class imbalances within the MCI category. The data is heavily skewed to the assault class with more than 50% of the observations pertaining to this class within MCI.

Table 5. Class imbalance

```
Class Distribution:
0    0.508825
2    0.212855
1    0.163048
3    0.078590
4    0.036682
Name: MCI_CATEGORY, dtype: float64
The dataset is imbalanced.
```

Brownlee (2021) states that the synthetic minority oversampling technique (SMOTE) effectively addresses class imbalance on classification datasets. Smote consists of oversampling the minority class without adding any new information to the model but synthetizing new examples from the existing ones in the dataset. Therefore, given the class imbalance in our dataset, the SMOTE technique is utilized in this project to get class-balanced training set and prevent our machine learning models from performing poorly (Table 6 and Figure 23).

Table 6. Class Balance

```
Class Distribution after SMOTE:
0    113292
4    113292
2    113292
3    113292
1    113292
Name: MCI_CATEGORY, dtype: int64
```

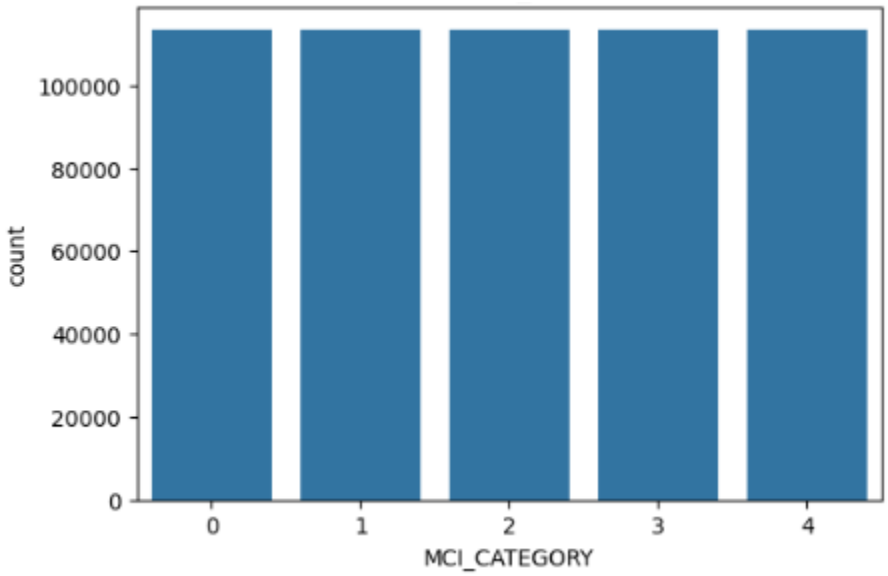


Figure 23. Balanced Target Variable in Training Set

At the outset, three classification algorithms were selected to assess crime prediction in Toronto: Decision Tree, KNN, and Naïve Bayes models. Initially, these models were trained on the designated training set and subsequently evaluated on the test set, without using any cross-validation. However, perfect scores and near-perfect scores of 100% accuracy were achieved with the Decision Tree and Naïve Bayes algorithms, respectively. Given that the accuracy score is calculated as the sum of correct classification divided by the total number of classifications, it is important to note that such results may not be entirely reliable due to potential overfitting, where the model may have memorized the training data rather than truly learning the underlying patterns. This can lead to poor performance when applied to unseen data.

To address this issue, the three classification models went through further evaluation utilizing a 10-fold cross-validation approach; however, the outcomes remained almost identical. Table 7 provides a summary of the overall accuracy given by each algorithm under both conditions—without cross-validation and with 10-fold cross-validation, respectively.

Table 7. Accuracy Results

Classifier	Method			
	Train Test Split		10-fold Cross Validation	
	Training Set	Test Set	Training Set	Test Set
Decision Tree	100.0%	100.0%	100.0%	100.0%
KNN	89.1%	73.5%	81.8%	73.5%
Naïve Bayes	99.5%	99.1%	99.4%	99.1%

While the accuracy score provides the proportion of correctly classified instances, achieving a perfect 100% accuracy can be an anomaly and may not necessarily reflect a trustworthy model. To address this, I adopted a distinct approach by initially dividing the dataset into training and test sets. This splitting from the very beginning allowed for a clear evaluation of the model's performance on completely separate data, eliminating the risk of overfitting. Despite this new approach, the accuracy results obtained for both the training and test sets remained quite similar to the previous findings.

Table 8 shows the time taken to train and evaluate the various algorithms using both, train test split and 10-fold cross validation methods.

Table 8. Summary of Model Implementation Time

Classifier	Implementation time	
	Train_Test_Split	10-fold Cross Validation
Decision Tree	3s	13 s
KNN	31m 51 s	43 m 22 s
Naïve Bayes	13 s	1 m 56 s

Performance Evaluation

The confusion matrices resulting from the models' evaluation can be seen in Appendix 3. Each confusion matrix shows the prediction classes in the columns and the actual classes in the rows. Since a multiclass problem is been evaluated, the values of the diagonal are the True Positives of the corresponding classes, and the off-diagonal values will be the errors. In that sense, the more zeroes or smaller the numbers on all cells but the diagonal, the better the classifier is performing.

As expected, the confusion matrices for the utilized models primarily yield values along the diagonal, signifying correct classifications. Only with KNN and Naïve Bayes models, some values appear off the diagonal.

The total number of test values of any class would be the sum of the corresponding row, which represents the True Positives (TP) plus the False Negatives (FN) for that class. Therefore, the total number of FN is the sum of values in the corresponding row, excluding the TP.

Similarly, the total number of False Positives (FP) for a class is the sum of values in the corresponding column, excluding the TP; while the total number of True Negatives (TN) for a certain class is the sum of all columns and rows, excluding the column and row corresponding to that class.

Appendix 4 shows the report of precision and recall metrics to evaluate the performance of each classifier. As it is expected, as the models appear to be highly accurate, precision and recall have also very good measures. It is interesting to note that computing precision and recall in a multiclass classification problem is not the same than calculating accuracy where we could get a general score of the algorithm performance, instead precision and recall needs to specify which class we are computing the precision and recall for.

Precision measures the proportion of TP predictions among all FP predictions. For example, the precision for the class "Assault" is calculated by dividing the number of correctly classified instances of "Assault" (found on the diagonal of the confusion matrix for the “Assault” class) by the sum of this number and the errors made when other classes—"Auto Theft," "Break and Enter," "Robbery," and "Theft Over"-- are misclassified as “Assault”. This calculation is represented by the equation below.

$$Precision_{Assault} = \frac{TP_{Assault}}{(TP_{Assault} + E_{Auto\ Theft/Assault} + E_{B\&E/Assault} + E_{Robbery/Assault} + E_{Theft\ Over/Assault})}$$

Recall, also known as the sensitivity, measures the proportion of TP predictions that were correctly classified by the algorithm. For example, precision for class "Assault" is calculated by dividing the number of correctly classified instances of "Assault" (found on the diagonal of the confusion matrix for the "Assault" class) by the sum of this number and the errors made when the instances are classified as other classes (such as "Auto Theft," "Break and Enter," "Robbery," and "Theft Over") when they are truly "Assault". This calculation is represented by the equation below.

$$Recall_{Assault} = \frac{TP_{Assault}}{(TP_{Assault} + E_{Assault/Auto\ Theft} + E_{Assault/B\&E} + E_{Assault/Robbery} + E_{Assault/Theft\ Over})}$$

Both precision and recall are important model evaluation metrics and it must be noted that it is not possible to maximize both these metrics at the same time without going through a trade-off, it means, one comes at the cost of another. Table 9 captures the precision and recall for the different classes for each of the algorithms implemented.

Table 9. Precision and Recall by Class

Decision Tree	Assault	Auto Theft	B & E	Robbery	Theft Over
Precision	1.00	1.00	1.00	1.00	1.00
Recall	1.00	1.00	1.00	1.00	1.00
KNN	**Assault**	**Auto Theft**	**B & E**	**Robbery**	**Theft Over**
Precision	0.93	0.63	0.59	0.46	0.25
Recall	0.87	0.63	0.63	0.54	0.30
Naïve Bayes	**Assault**	**Auto Theft**	**B & E**	**Robbery**	**Theft Over**
Precision	1.00	0.99	1.00	0.90	1.00
Recall	0.98	1.00	1.00	1.00	0.97

Among the three models implemented, the KNN classifier demonstrates acceptable performance for our objectives. All classes exhibit good precision and recall scores, except for "Theft Over," for which none of the metrics appear to be satisfactory.

Revisiting Feature Selection

Given the inconsistent performance of the models, I revisited the feature selection process. This time, I employed the SelectKBest technique from the scikit-learn library, which identifies the top k features based on statistical tests. Specifically, I selected 10 features considered most relevant (Table 10) and applied this selection exclusively to the training set to avoid overfitting.

Subsequently, I reprocessed the dataset, retaining only the selected features while discarding the rest. To address class imbalance, SMOTE was once again utilized to balance the training set.

Following this preprocessing step, classification algorithms were trained using 10-fold cross-validation.

Table 10. Top 10 Features Based on SelectKBest

```
X
REPORT_YEAR
OCC_YEAR
OCC_HOUR
LOCATION_TYPE
PREMISES_TYPE
UCR_CODE
UCR_EXT
OFFENCE
HOOD_140
```

Table 11 shows the resulting accuracy scores for the new model implementation using SelectKBest feature selection technique. The results for Decision Tree and Naïve Bayes algorithms are basically the same than the ones gotten when used the information gain technique, while KNN algorithm has yielded higher scores, climbing from 73.5% to 93.5%.

Table 11. Accuracy Results – Feature Selection Revisited

Classifier	10-fold Cross Validation	
	Training Set	Test Set
Decision Tree	100.0%	100.0%
KNN	97.9%	93.6%
Naïve Bayes	100.0%	100.0%

Conclusions

- The accuracy scores for the three selected machine learning classification algorithms raise concerns about their performance. The Decision Tree classifier demonstrates consistency, achieving cross-validation scores across all folds on the training set, with a corresponding accuracy on the test set. The KNN model exhibits strong performance, with mean cross-

validation scores averaging high on the training set and achieving a notable accuracy on the test set. The Naïve Bayes classifier, similar to the Decision Tree, achieves cross-validation scores and test set accuracy. The occurrence of models with 100% accuracy is unusual and may indicate potential issues such as overfitting or data leakage. Based on this, it appears that KNN emerges as the most reliable model for predicting crime occurrences in Toronto.

- After the analysis of two applied feature selection techniques, the variables related to location patterns, such as the type of premises in which the crime occurred, the structure of the neighborhood where the crimes took place, and the geographic location, were identified as the most influential features for the classification models.
- Among the algorithms examined in this project, the KNN method stands out as particularly promising and holds the potential to forecast crime in Toronto. All the five categories of major crime indicators may be accurate predictive, except for the "Theft Over" category, which is the act of stealing property in excess of $5,000.
- These findings could heighten public awareness of high-risk areas in Toronto, aiding efforts by the Toronto Police Service to anticipate future crimes at specific locations and times.

Project Challenges and Future Work.

- In this project, only information gain and SelectKBest techniques for feature selection were applied. In the future, other filter-based, wrapper-based, and embedded techniques could be used to analyze the performance of the selected algorithms.

- Likewise, only SMOTE was used to balance the training set but in future work, other under sampling methods and oversampling methods could be applied and see how these perform on the dataset.
- Unfortunately, I was only able to use the label encoder function to deal with the categorical variables, I tried using get_dummies, but the google colab file crashed due to lack of RAM.
- In this project, data scaling was not utilized. Exploring alternative data transformation methods in future iterations could be beneficial.
- Further investigation is warranted to understand the reasons behind the performance metrics of 100%. I have tried many more ways than those reported in this report to resolve this situation, however, they have all given similar results unfortunately.

References

Almanie, T., Mirza, R., & Lor, E. (2015). Crime Prediction Based On Crime Types And Using Spatial And Temporal Criminal Hotspots. *International Journal of Data Mining & Knowledge Management Process*, 5(4), 1-20. DOI:10.5121/ijdkp.2015.5401

Brownlee, J. (2021, March 16). *Smote for imbalanced classification with python. MachineLearningMastery.com.* https://machinelearningmastery.com/smote-oversampling-for-imbalanced-classification/

Esfahani, H. N. and Esfahani, Z. N. (2023). *Exploring crime rate trends and forecasting future patterns in Toronto city using police mci data and deep learning*. https://doi.org/10.21203/rs.3.rs-3806294/v1

Gupta, A. (2023, December 21). *Feature selection techniques in Machine Learning (updated 2024)*. Analytics Vidhya. https://www.analyticsvidhya.com/blog/2020/10/feature-selection-techniques-in-machine-learning/

Hussain, F. S., & Aljuboori, A. F. (2022). A Crime Data Analysis of Prediction Based on Classification Approaches. *Baghdad Science Journal*, 19(5), 1073. https://doi.org/10.21123/bsj.2022.6310

Kumar, A., Verma, A., Shinde, G., Sukhdeve, Y. & Lal, N. (2020). *Crime Prediction Using K-Nearest Neighboring Algorithm*, 2020 International Conference on Emerging Trends in Information Technology and Engineering (ic-ETITE), Vellore, India, 1-4, 10.1109/ic-ETITE47903.2020.155.

Ontario (2013). *Open Government Licence - Ontario*. (Published 2013, June 18. Updated 2023, May 10). https://www.ontario.ca/page/open-government-licence-ontario

Saeed, R. & Abdulmohsin, H. (2023). A study on predicting crime rates through machine learning and data mining using text, *Intelligent Systems*, 32(1), 20220223. https://doi.org/10.1515/jisys-2022-0223

Safat, W., Asghar, S., & Gillani, S.A. (2021). Empirical Analysis for Crime Prediction and Forecasting Using Machine Learning and Deep Learning Techniques. *IEEE Access*, 9, 70080-70094.

The Chang School of Continuing Education (Fall Term, 2023). *Module 10.1 – Classification vs Clustering*. Module 10. Statistical Learning Methods. CMTH642 - Data Analytics: Advanced Methods.

Toronto Police Service (2024, January 22). *Major Crime Indicators Open Data*. Toronto Police Service - Public Safety Data Portal. Retrieved January 22, 2024, from https://data.torontopolice.on.ca/datasets/TorontoPS::major-crime-indicators-open-data/about

Vaquero Barnadas, M. (2016, November 14). *Machine learning applied to crime prediction*. Handle Proxy. http://hdl.handle.net/2117/96580

Appendix

Appendix 1. Major Crime Indicators (MCI) Glossary

MCI	Definition
Assault	The direct or indirect application of force to another person, or the attempt or threat to apply force to another person, without that person's consent.
Auto Theft	The act of taking another person's vehicle (not including attempts).
Break and Enter	The act of entering a place with the intent to commit an indictable offence therein.
Robbery	The act of taking property from another person or business by the use of force or intimidation in the presence of the victim.
Theft Over	The act of stealing property in excess of $5,000 (auto theft is excluded).

Source: Toronto Police Service (2024)

Appendix 2. Attributes Dictionary of MCIs Dataset[1]

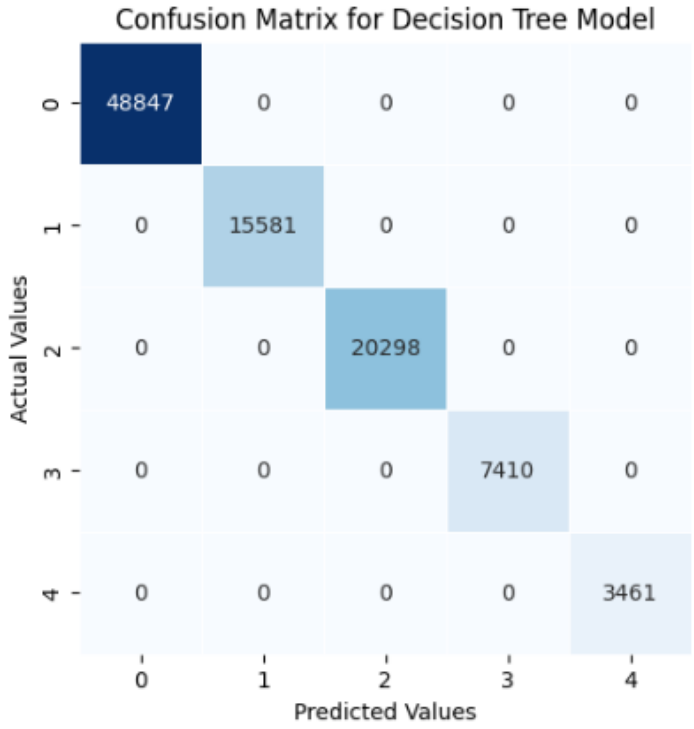
Field	Attribute Name	Description
1	EVENT_UNIQUE_ID	Offence Number
2	REPORT_DATE	Date Offence was Reported (time is displayed in UTC format when downloaded as a CSV)
3	OCC_DATE	Date Offence Occurred (time is displayed in UTC format when downloaded as a CSV)
4	REPORT_YEAR	Year Offence was Reported
5	REPORT_MONTH	Month Offence was Reported

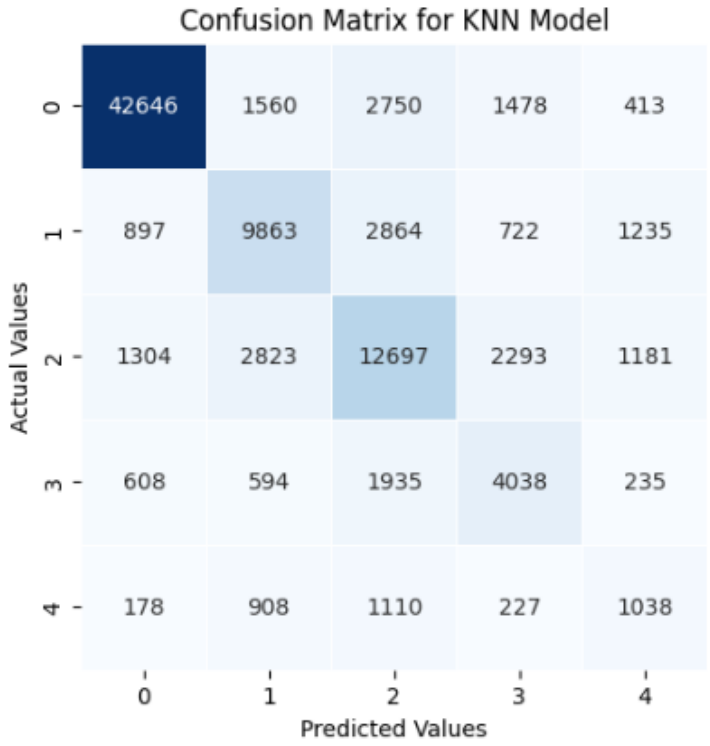
[1] Dataset description retrieved from Toronto Police Service (2024) did not include attributes X, Y and OBJECTID.

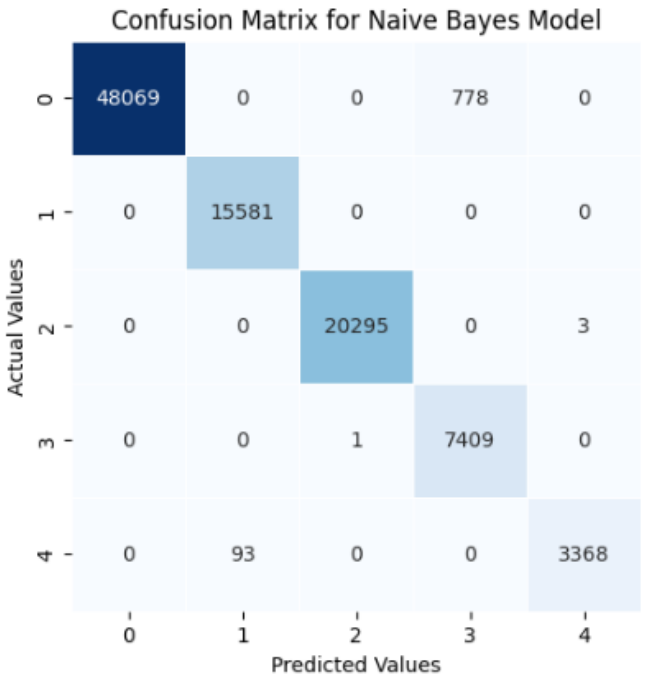
6	REPORT_DAY	Day of the Month Offence was Reported
7	REPORT_DOY	Day of the Year Offence was Reported
8	REPORT_DOW	Day of the Week Offence was Reported
9	REPORT_HOUR	Hour Offence was Reported
10	OCC_YEAR	Year Offence Occurred
11	OCC_MONTH	Month Offence Occurred
12	OCC_DAY	Day of the Month Offence Occurred
13	OCC_DOY	Day of the Year Offence Occurred
14	OCC_DOW	Day of the Week Offence Occurred
15	OCC_HOUR	Hour Offence Occurred
16	DIVISION	Police Division where Offence Occurred
17	LOCATION_TYPE	Location Type of Offence
18	PREMISES_TYPE	Premises Type of Offence
19	UCR_CODE	UCR Code for Offence
20	UCR_EXT	UCR Extension for Offence
21	OFFENCE	Title of Offence
22	MCI_CATEGORY	MCI Category of Occurrence
23	HOOD_158	Identifier of Neighbourhood using City of Toronto's new 158 neighbourhood structure
24	NEIGHBOURHOOD_158	Name of Neighbourhood using City of Toronto's new 158 neighbourhood structure
25	HOOD_140	Identifier of Neighbourhood using City of Toronto's old 140 neighbourhood structure
26	NEIGHBOURHOOD_140	Name of Neighbourhood using City of Toronto's old 140 neighbourhood structure
27	LONG_WGS84	Longitude Coordinates (Offset to nearest intersection)
28	LAT_WGS84	Latitude Coordinates (Offset to nearest intersection)

Retrieved from: Toronto Police Service (2024)

Appendix 3. Confusion Matrices







Appendix 4. Precision, Recall and f1-scores

Metrics for Decision Tree Algorithm:

	precision	recall	f1-score	support
0	1.00	1.00	1.00	48847
1	1.00	1.00	1.00	15581
2	1.00	1.00	1.00	20298
3	1.00	1.00	1.00	7410
4	1.00	1.00	1.00	3461
accuracy			1.00	95597
macro avg	1.00	1.00	1.00	95597
weighted avg	1.00	1.00	1.00	95597

Metrics for KNN Algorithm:

	precision	recall	f1-score	support
0	0.93	0.87	0.90	48847
1	0.63	0.63	0.63	15581
2	0.59	0.63	0.61	20298
3	0.46	0.54	0.50	7410
4	0.25	0.30	0.27	3461
accuracy			0.74	95597
macro avg	0.57	0.60	0.58	95597
weighted avg	0.75	0.74	0.74	95597

Metrics for Naive Bayes Algorithm:

	precision	recall	f1-score	support
0	1.00	0.98	0.99	48847
1	0.99	1.00	1.00	15581
2	1.00	1.00	1.00	20298
3	0.90	1.00	0.95	7410
4	1.00	0.97	0.99	3461
accuracy			0.99	95597
macro avg	0.98	0.99	0.98	95597
weighted avg	0.99	0.99	0.99	95597

Applying 10-fold Cross Validation

Classifier: Decision Tree

	precision	recall	f1-score	support
0	1.00	1.00	1.00	48847
1	1.00	1.00	1.00	15581
2	1.00	1.00	1.00	20298
3	1.00	1.00	1.00	7410
4	1.00	1.00	1.00	3461
accuracy			1.00	95597
macro avg	1.00	1.00	1.00	95597
weighted avg	1.00	1.00	1.00	95597

Classifier: KNN

	precision	recall	f1-score	support
0	0.93	0.87	0.90	48847
1	0.63	0.63	0.63	15581
2	0.59	0.63	0.61	20298
3	0.46	0.54	0.50	7410
4	0.25	0.30	0.27	3461
accuracy			0.74	95597
macro avg	0.57	0.60	0.58	95597
weighted avg	0.75	0.74	0.74	95597

Classifier: Naive Bayes

	precision	recall	f1-score	support
0	1.00	0.98	0.99	48847
1	0.99	1.00	1.00	15581
2	1.00	1.00	1.00	20298
3	0.90	1.00	0.95	7410
4	1.00	0.97	0.99	3461
accuracy			0.99	95597
macro avg	0.98	0.99	0.98	95597
weighted avg	0.99	0.99	0.99	95597