Chicago Crime Data Analysis

ABSTRACT

This study focuses on finding patterns and improving prediction abilities via analyzing the Chicago crime data thoroughly using a variety of machine learning models. The dataset contains a variety of variables, such as time, place, and types of crimes. The project tackles issues like handling categorical variables and model optimization by carefully preprocessing and using models like Logistic Regression, Decision Trees, Random Forest, Naive Bayes, k-nearest Neighbors, and Neural Networks. The evaluation criteria that direct the comparison of model performances include recall, accuracy, precision, and F1-score. The conclusions drawn help create a more complex picture of Chicago's crime dynamics, which helps law enforcement and urban security agencies make wise decisions. The effective use of data analytics for public safety and crime prevention in urban settings is demonstrated by this study.

Keywords – Random Forest, Naïve Bayes, Decision Tress, Neural Networks, Logistic Regression, CLEAR (Citizen Law Enforcement Analysis and Reporting)

I. OVERVIEW

A. Problem Statement

In urban environments, crime is a multifaceted challenge that significantly impacts public safety and the quality of life. Chicago, being one of the major cities in the United States, presents a diverse and complex pattern of criminal activities. This study aims to delve into the intricate landscape of crime in Chicago, with a focus on uncovering temporal patterns, spatial distributions, and predicting crime types. Such analysis is vital for informed decision-making in law enforcement and urban safety planning.

II. PROPOSED METHODOLOGY/APPROACH

A. Data Gathering

For this project, we have utilized the dataset Chicago Police Departments CLEAR (Citizen Law Enforcement Analysis and Reporting) [1] from their official website and we have used the dataset of 2020 to 2022 as our baseline dataset. Additionally, we have also accessed the dataset from 2001 onwards to 2023 and done Exploratory Data Analysis on the entire dataset. Our initial goal was to carry out an analysis of the data for each year and compare our findings.

However, to narrow down our problem statement we decided to only use the 2020 to 2022 dataset as our sample for research study in this project. This approach has allowed us to simplify our problem statement and make our research analysis goal easier.

B. Data Preprocessing – Pipeline Details

In order to prepare the dataset for analysis, several steps were taken, such as ensuring data consistency by assigning a proper data type to each data point, handling missing values by imputing them with either a fixed value or a statistical measure such as mean, median, or mode, and dealing with invalid entries. Moreover, we made extensive effort to remove redundant or unnecessary columns. We also used techniques such as outlier detection and hot encoding to refine the data further and prepare it for analysis.

C. Data Cleaning

The dataset had a lot of variances in terms of data needed for research, hence we spent a great deal of time cleaning the data. Here are some of the methods we employed during this process:

- The datetime stamp was combined in 1 column and we had to separate out the content from the date to month, day, and year. Additionally, the hours were also separate but were not required.
- We dropped the unnecessary columns to reduce the dimensionality of the data. For example, we removed the latitude and longitude coordinates as we had the location, but each coordinate was not required for this research study.
- We also had to manually specify the column names as the original file contained long questions as the column names. This process was trivial yet cumbersome.

III. DATA ANALYSIS

A. Summary Statistics and Visualization

Since most of our data was categorical, it did not make sense to find statistical summaries like mean values, or variance. Therefore, we resorted to segmenting the data into different rows (grouping into different groups or

categories) and analyze the dataset using EDA (Exploratory Data Analysis).

The methods employed in this study include using bar plots to analyze the relationships between various factors and the response variable, such as type of crimes, community areas where crime are committed and distribution of crime over the years. To observe any trends in a specific group of people, colored bar plots are drawn by label with categorical variables. We will take a look at some of the plots which display important relationships in the subsequent sections.

Plotting libraries such as ggplot, and tree map was used to visualize the results of exploratory data analysis (EDA) and interpret findings. Additionally, histograms, heatmaps, and bar charts were used appropriately to map the relationships between predictors and the variables.

B. Exploratory Data Analysis

To visualize the dataset, we started off with the different finding out the top ten types of crimes committed in Chicago over the years and what type of crime are most common. As we can see from the pie chart below: theft and battery contribute the majority of area of the pie chart while criminal damage and narcotics are other categories that also take up a considerable share of the crimes committed.

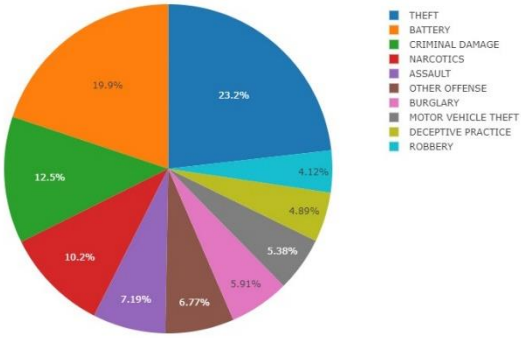

Figure 1 - Top 10 crimes committed in Chicago

Next up we try to visualize the different areas where crimes are committed majorly. We build a tree map for this part to identify the crime density by community area. The results are summarized in the Figure below and we can see that Austin, Near North Side, South Shore, West Town are some of the most prominent areas.

Figure 2 - Tree Map of Crime Density by Community Areas

Furthermore, we have also identified the domestic crimes in Chicago from our dataset and we see from the figure below that we have only 17.3% of domestic crimes in Chicago over the years.

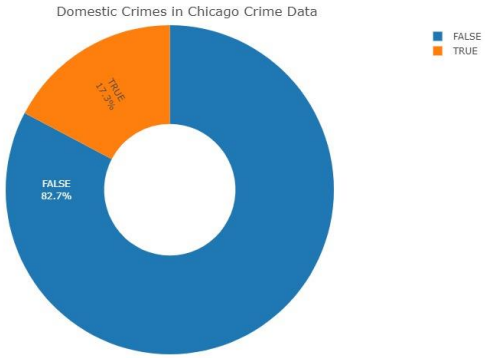

Figure 3 - Domestic Crime Percentage

Next, we have the percentage distribution of arrests in Chicago based on our dataset. We see for the number of crimes committed only one quarter (around 25.5%) of those offenders are actually arrested which is a considerably low percentage if we look in terms of the number of crimes committed.

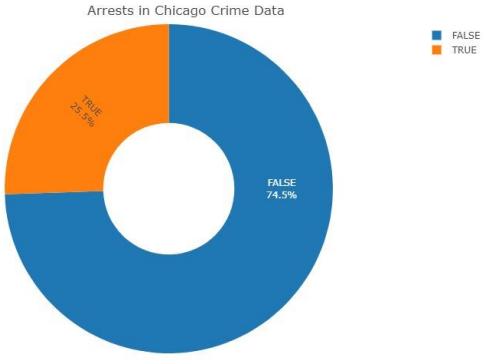

Figure 4 - Arrests Rate in Chicago

Next up we have correlation matrix and heatmap to visualize the relationships between different features in the dataset. The correlation coefficients are shown in the table, with red representing positive correlation and blue representing negative correlation. The stronger the color, the stronger the correlation.

Some of the key observations from the data are:

- Arrests are more likely to occur on weekends and at night. This is supported by the positive correlations between Arrest and Day and Hour.
- Domestic arrests are more likely to occur in the home. This is supported by the positive correlation between Domestic and Location Description.
- There is a strong correlation between the Primary Type of arrest and whether or not it is domestic. This suggests that the type of crime is often a good predictor of whether or not it is domestic violence.

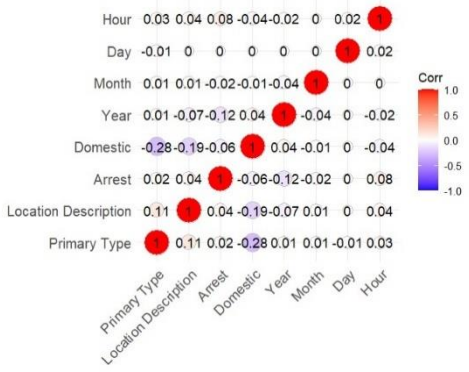

Figure 5 - Correlation Matrix b/w features

C. *Feature Extraction*

For the purpose of predictive modeling, significant features were extracted from the dataset:

- Temporal Features: Variables such as the time of day, day of the week, and month were derived from the date and time data to capture temporal patterns in crime occurrences.
- Spatial Features: Geographic coordinates were utilized as key spatial features, enabling the analysis of crime locations and their distribution across the city.

The data analysis phase provided critical insights into the nature and characteristics of crime in Chicago. It laid the groundwork for the subsequent predictive modeling, guiding the selection of features and the formulation of the model.

IV. MODEL TRAINING

A. *Model Selection and Initialization*

Our project explored various machine learning algorithms to predict crime types in Chicago based on temporal and spatial factors. The models considered include Logistic Regression, Decision Tree, Random Forest, Naïve Bayes (Multinomial), K-nearest Neighbors, Support Vector Machine, and Neural Network (Multilayer Perceptron). Each model was selected for its potential effectiveness in classification tasks involving categorical data.

B. *Data Preprocessing*

The preprocessing steps were crucial to ensure the compatibility of the data with our models. This included feature scaling and encoding categorical variables like 'Primary Type' and 'Location Description.' The dataset was segmented by year, with dimensionality reduction techniques applied for a more focused analysis.

C. *Model Training and Evaluation*

The training process involved stratified K-Fold cross-validation, chosen for its effectiveness in handling imbalanced datasets. Each model's performance was evaluated based on accuracy scores across different folds. The average accuracy scores were compared to guide the selection of the most suitable model.

D. *Model Comparison/Selection*

The following models were considered for classification, and their average accuracy scores were compared to guide the selection of the most suitable model:

Model	Fold 1 Accuracy	Fold 2 Accuracy	Fold 3 Accuracy	Average Accuracy
Multinomial Logistic Regression	0.43945	0.43849	0.43879	0.43891
Decision Tree	0.41687	0.41549	0.41794	0.41676
Random Forest	0.43793	0.43822	0.43912	0.43842
Naïve Bayes (Multinomial)	0.42179	0.42131	0.42224	0.42178

K-nearest Neighbors	0.40243	0.40101	0.40308	0.40217
Neural Network (MLP)	0.53047	0.52957	0.55192	0.53065

Table 1 - Model Accuracies

V. MODEL VALIDATION

A. *Hyperparameter Tuning and Model Selection*

Hyperparameter tuning was particularly applied to the Neural Network model, exploring various configurations to maximize accuracy. The best model configuration was identified through a grid search over a range of hyperparameters.

B. *Hyperparameter Search*

A grid search was conducted over a range of hyperparameters, including:

- Hidden layer sizes
- Activation functions
- Solvers
- Alpha (L2 regularization term)
- Learning rate

```
Fitting 3 folds for each of 96 candidates, totalling 288 fits
[CV] END activation=tanh, alpha=0.0001, hidden_layer_sizes=(50,), learning_rate=constant, solver=sgd; total time= 6.1min
[CV] END activation=tanh, alpha=0.0001, hidden_layer_sizes=(50,), learning_rate=adaptive, solver=sgd; total time= 6.3min
[CV] END activation=tanh, alpha=0.0001, hidden_layer_sizes=(50,), learning_rate=adaptive, solver=sgd; total time= 6.5min
[CV] END activation=tanh, alpha=0.0001, hidden_layer_sizes=(50,), learning_rate=constant, solver=sgd; total time= 6.8min
[CV] END activation=tanh, alpha=0.0001, hidden_layer_sizes=(50,), learning_rate=constant, solver=adam; total time= 7.8min
[CV] END activation=tanh, alpha=0.0001, hidden_layer_sizes=(50,), learning_rate=adaptive, solver=adam; total time= 8.0min
[CV] END activation=tanh, alpha=0.0001, hidden_layer_sizes=(50,), learning_rate=adaptive, solver=adam; total time= 8.1min
[CV] END activation=tanh, alpha=0.0001, hidden_layer_sizes=(50,), learning_rate=constant, solver=adam; total time= 8.4min
[CV] END activation=tanh, alpha=0.0001, hidden_layer_sizes=(50,), learning_rate=constant, solver=sgd; total time= 9.7min
[CV] END activation=tanh, alpha=0.0001, hidden_layer_sizes=(50,), learning_rate=adaptive, solver=sgd; total time=10.1min
[CV] END activation=tanh, alpha=0.0001, hidden_layer_sizes=(50,), learning_rate=constant, solver=adam; total time=12.2min
[CV] END activation=tanh, alpha=0.0001, hidden_layer_sizes=(100,), learning_rate=constant, solver=sgd; total time= 7.9min
[CV] END activation=tanh, alpha=0.0001, hidden_layer_sizes=(100,), learning_rate=constant, solver=sgd; total time= 8.7min
[CV] END activation=tanh, alpha=0.0001, hidden_layer_sizes=(100,), learning_rate=constant, solver=sgd; total time= 9.1min
[CV] END activation=tanh, alpha=0.0001, hidden_layer_sizes=(50,), learning_rate=adaptive, solver=adam; total time=15.4min
[CV] END activation=tanh, alpha=0.0001, hidden_layer_sizes=(100,), learning_rate=adaptive, solver=sgd; total time=10.1min
[CV] END activation=tanh, alpha=0.0001, hidden_layer_sizes=(100,), learning_rate=constant, solver=adam; total time=11.1min
[CV] END activation=tanh, alpha=0.0001, hidden_layer_sizes=(100,), learning_rate=adaptive, solver=sgd; total time=11.1min
[CV] END activation=tanh, alpha=0.0001, hidden_layer_sizes=(100,), learning_rate=adaptive, solver=sgd; total time=10.4min
[CV] END activation=tanh, alpha=0.0001, hidden_layer_sizes=(100,), learning_rate=constant, solver=adam; total time=13.7min
[CV] END activation=tanh, alpha=0.0001, hidden_layer_sizes=(100,), learning_rate=constant, solver=adam; total time=13.3min
[CV] END activation=tanh, alpha=0.0001, hidden_layer_sizes=(50, 50), learning_rate=constant, solver=adam; total time= 5.5min
[CV] END activation=tanh, alpha=0.0001, hidden_layer_sizes=(100,), learning_rate=adaptive, solver=adam; total time=15.3min
```

Figure 6 - Code Snippet 1

C. *Best Model Selection*

The hyperparameter tuning process identified the best configuration for the Neural Network model, achieving an accuracy of 0.65103, and the corresponding parameters were:

- Activation function: ReLU
- Regularization Term (Alpha): 0.001
- Hidden Layer Sizes: (50, 50)
- Learning Rate: Adaptive

D. *Conclusion*

The validation phase affirmed the model's effectiveness in predicting the primary type of crime in Chicago, while also highlighting areas for improvement. The insights gained from this process are instrumental for refining the model and guiding its deployment in practical applications.

VI. CONCLUSION

A. *Positive Results:*

Our analysis started with models that were marginally better than chance, achieving an accuracy of around 45% against a baseline of 10% for random guessing across 10 crime categories. Through systematic tuning, including grid search and hyperparameter adjustments, we improved the Neural Network model's accuracy by approximately 10%. This underscores the capabilities of machine learning techniques in enhancing our understanding of crime data patterns.

B. *Negative Results:*

However, our models struggled to accurately predict the primary crime type. The correlation analysis revealed weak relationships between the features we used and the crime categories, suggesting that key predictive variables may be missing from our study.

C. *Recommendations for Future Work:*

To refine our predictive accuracy, we recommend incorporating additional data points into the dataset. Variables such as weather patterns, socioeconomic metrics, and demographic details are known to influence crime and could provide the necessary depth for more accurate predictions.

D. *Caveats and Cautions:*

It's important to note that the predictive models we've developed are based on historical data and may not account for all the complexities of criminal behavior. The effectiveness of these models can be influenced by data quality, the relevance of the features chosen, and changes in societal patterns over time. Furthermore, the ethical implications of using demographic data in predictive policing should be carefully considered to avoid potential biases. Users of these models should apply them judiciously and remain aware of these limitations.

VII. SOURCE CODE & DATA

- Data: Chicago Data Portal
- Source Code: GitHub
- Reference Data: Kaggle

VIII. REFERENCES

[1] Chicago Police Department, "Chicago Data Portal," 1 December 2023. [Online]. Available: https://data.cityofchicago.org/Public-Safety/Crimes-2001-to-Present/ijzp-q8t2.

[2] Kaggle, "Crime data analysis," 03 September 2018. [Online]. Available: https://www.kaggle.com/code/sevgisarac/crime-data-analysis/input.

[3] ahillard, "Chicago-Crime-Data-Analysis," 30 November 2016. [Online]. Available: https://github.com/ahillard/Chicago-Crime-Data-Analysis.

[4] Micheal Minn, "Crime Point Data Analysis in R," October 2021. [Online]. Available: https://michaelminn.net/tutorials/r-crime/index.html.