Project Report

Crime Prediction in Syracuse

PROJECT OVERVIEW

The Big Data Analytics project focusing on Crime Prediction in Syracuse is a comprehensive endeavor aimed at harnessing large-scale data to forecast and analyze crime patterns within the city. The primary objective is to employ advanced data analytics techniques on diverse datasets encompassing historical crime records, socio-economic indicators, demographic information, weather patterns, and geographical features. The project will be initiated with meticulous data collection and preparation, ensuring the cleanliness and compatibility of the datasets for subsequent analysis and Exploratory Data Analysis (EDA). Feature engineering techniques will be implemented to identify crucial factors influencing crime rates, which will be used to augment predictive models. Machine learning algorithms such as Random Forest, Decision Trees, Gradient Boosted Trees are being used in this project to develop robust models capable of predicting future crime occurrences in Syracuse. Ethical considerations regarding privacy, bias mitigation, and fairness in predictions will be paramount throughout the project's lifecycle. The goal is to provide actionable insights for law enforcement to proactively address and reduce crime rates, thereby enhancing public safety in Syracuse.

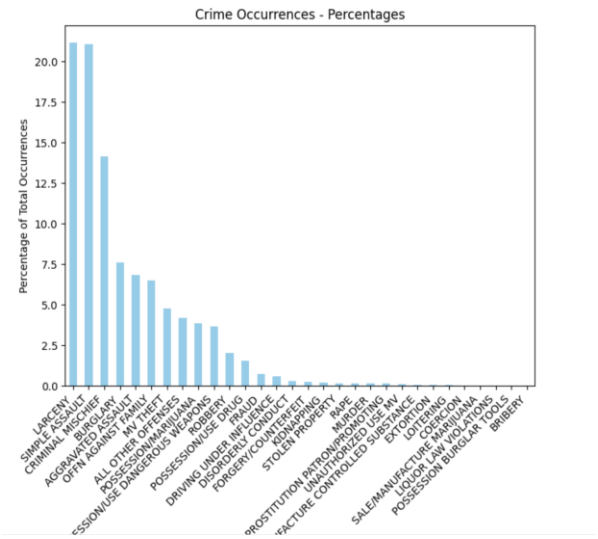
DATASET DESCRIPTION

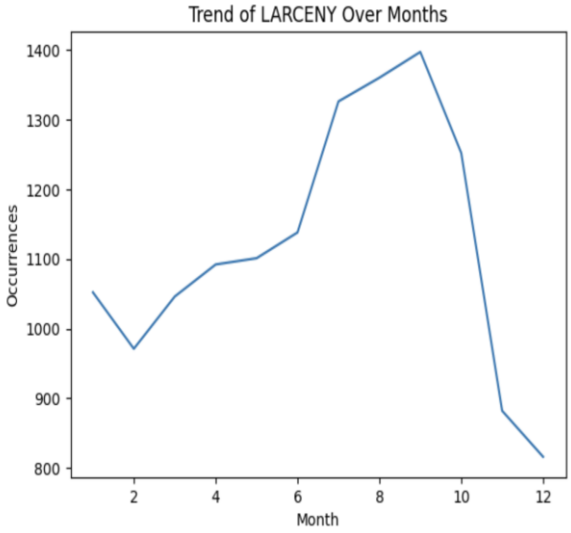
The Syracuse crime data is obtained from the City of Syracuse. The data is for 5 years from 2019 to 2023. This crime data is divided into Part 1 and Part 2 offense. Part 1 contains Criminal Homicide, Forcible rape, Robbery, Aggravated assault, Burglary, Larceny-theft and Motor Vehicle Theft whereas Part 2 includes all the offenses – Kidnapping, Extorsion, Simple Assault, Stolen Property, Bribery, Loitering etc. The following figure gives the number of columns and their description.

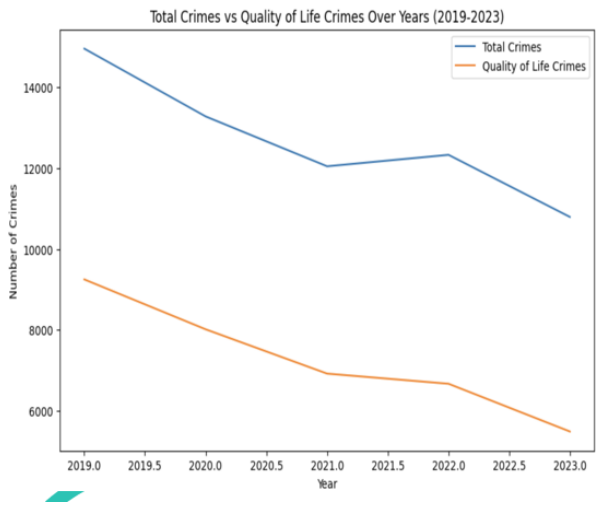
ObjectID: Unique Id to each record.
DateEnd: Date that the crime was reported. It could have happened earlier. This is in the format of DD-MON-YY (Ex. 01-Jan-22).
Time start and time end: Listed in military time (2400) - Burglaries and larcenies are often a time frame.
Address: Where the crime occurred. All addresses are in the 100's because the Syracuse Police Department allows privacy for residents and only lists the block number.
Code Defined: Offense names are listed as crime categories group for ease of understanding. There may have been other offenses also, but the one displayed is the highest Unified Crime Reporting (UCR) category.
Arrest: Means that there was an arrest, but not necessarily for that crime.
Larceny Code: Indicates the type of larceny (Example: From Building or From Motor Vehicle).

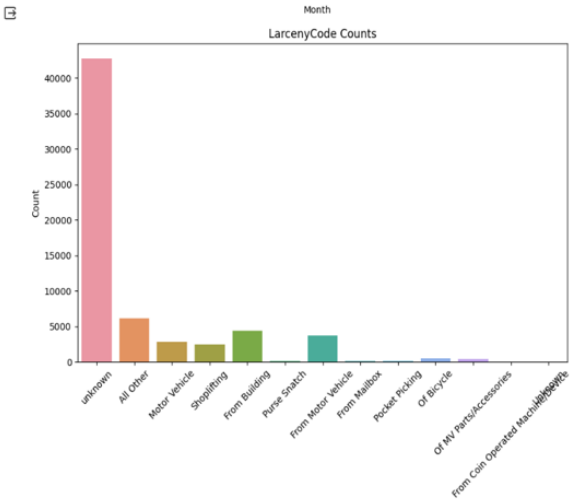
The 5 years (2019-2023) crime data is combined into a dataset, including Part 1 and Part 2 crime data. Upon combining, we got 63,449 records of the crime events and 11 variables.

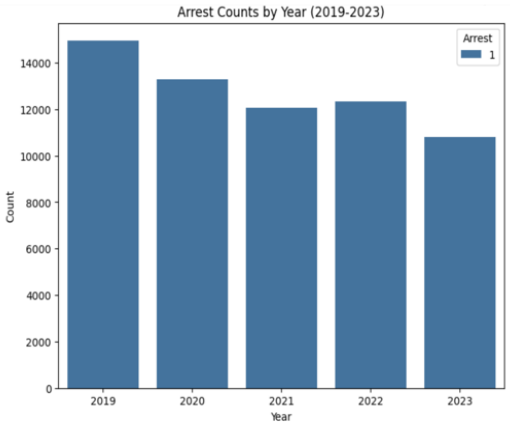
EXPLORATORY DATA ANALYSIS

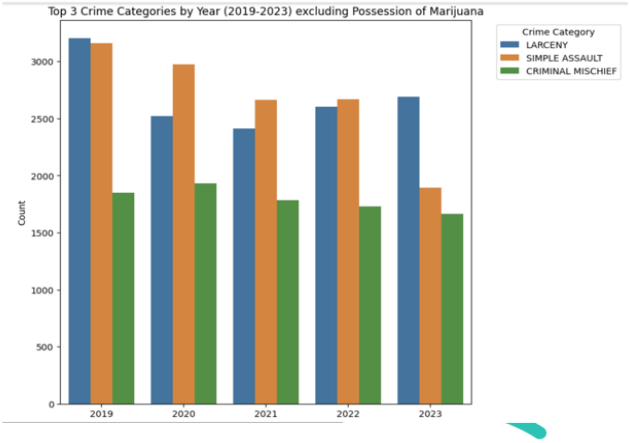












PREDICTION AND INFERENCE GOALS

1. **Crime Rate Prediction:** Overall crime rate in Syracuse for the coming year(s) based on historical data. Predicting the crime that is taking place in a particular location.
2. **Arrest Probability:** Create a model to predict the likelihood of an arrest occurring given a reported crime, considering factors such as time, location, and crime category. This can assist in resource allocation and investigation prioritization.
3. **Quality of Life Impact Prediction:** Predict the impact of crimes on the quality of life in Syracuse, considering factors like the type and frequency of crimes. This can inform community safety efforts.
4. **Type of Larceny Prediction:** Predict the type of larceny (e.g., from a building or from a motor vehicle) based on historical data. This information can be used to implement preventive measures.

DATA PRE-PROCESSING and FEATURE ENGINEERING

Initially, we addressed missing values within the dataset by inputting them appropriately. For numerical columns Arrest, it replaces NULL values. For categorical columns CODE_DEFINED, LarcenyCode, it substitutes NULL values with a default label ('unknown'). StringIndexer is utilized to convert categorical string columns such as CODE_DEFINED and LarcenyCode into numerical indices. OneHotEncoder further processes these indexed columns into sparse binary vectors, known as one-hot encoded vectors. These transformations are performed to prepare the categorical variables for inclusion in our ML models. Finally, a VectorAssembler merges the one-hot encoded categorical features with the numerical features 'Arrest' into a single feature vector named 'features'. This 'features' vector serves as the input for machine learning algorithms, ensuring compatibility and enabling the model to understand the data's characteristics and patterns effectively. The overall process establishes a structured data format ready for subsequent model training and evaluation within a machine learning pipeline.

SUMMARY OF METHODS

We have both regression and classification models in our project to predict the various variables to predict the crime.

1. Random Forest

We used Random Forest Model initially to predict the number of occurrences of crime based on the location. This model is used for its ability to handle large datasets and provide accurate predictions. Using this model, we are predicting the highest crime occurring in a particular location. ‘ADDRESS’ column contains the street names, in order to secure the privacy of the residents of Syracuse, the street names are added with 100 as a street number. Initially, it removes street numbers from the ‘ADDRESS’ column using regular expressions to create a new column ‘streetName’ free from these numerical values, ensuring that only street names remain. Next, the process involves tokenization, which splits the street names into individual words using the Tokenizer function. Following tokenization, the HashingTF (Hashing Term Frequency) method is applied to convert these tokenized words into numeric feature vectors for each address. The HashingTF method hashes words into numerical indices and computes their term frequencies in the document to form the feature vectors, set here to a dimension of 100. Subsequently, a Random Forest Regressor model is employed, taking these newly transformed 'features' as input and aiming to predict the 'Arrest' label. The model also calculates the sum of predicted values for each unique address, grouping the data by ‘ADDRESS’, and then sorts the results in descending order based on the total sum of predictions.

```
Row(ADDRESS='1 DESTINY USA DR', sum(prediction)=2062.0)
```

This allows for the identification of the location with the highest predicted values from the model, potentially indicating areas more prone to criminal activities based on the learned patterns from the input street names. Here we can see that Destiny USA has the highest crime. Based on the highest crime predicted, the model is also used to print the occurrence of crime.

```
The crime happened at 700 N TOWNSEND ST from 5:00 PM to 1:32 PM with a predicted count of 1.0.
```

2. Logistic Regression – Arrest Probability and Quality of Life prediction

This model is used to comprehensively evaluate the logistic regression model's performance in predicting the ‘Arrest’ and ‘QualityOfLife’. Initially, model is trained using the 'training_df' dataset and applied to the 'test_df' dataset to generate predictions. Two evaluators are used for performance assessment. The Binary Classification Evaluator computes the Area Under the Receiver Operating Characteristic (ROC) Curve, assessing the model's ability to distinguish between positive and negative classes (‘Arrest’ and ‘QualityOfLife’). The Multiclass Classification Evaluator is then utilized to calculate the accuracy of the model in predicting the 'Arrest' class labels, providing an overall measure of correctness. Next, a Multiclass Classification

Evaluator is instantiated with the parameter 'metricName' set to 'f1' for computing the F1-score, precision, and recall. The results are as follows.

```
Area Under ROC Curve: 1.0
Accuracy: 0.5774868699537509
```

```
Precision: 1.0
Recall: 0.5774868699537509
F1-score: 0.7321606042536275
```

3. Linear Regression

The Linear Regression model is employed for its simplicity and efficiency in predicting a continuous dependent variable based on independent variables. It's used here to predict the 'QualityOfLife' variable, likely a binary indicator of how crime affects the life of Syracuse residents. We have also used it to predict the 'Arrest' probability. Following are the results of QualityOfLife prediction.

```
+------------------+
|               mse|
+------------------+
|0.24421816791358583|
+------------------+
```

```
Root Mean Squared Error (RMSE): 0.4941843460831047
R-squared (R²): -2.9740369287711843e-05
```

From the model results, the MSE value is approximately 0.2442 and RMSE is 0.4942 and the R^2 value reported here is approximately -2.974e-05, which is very close to zero. This result indicates that the linear regression model did not perform well for this dataset, hence the model is not fit for the data. Hence, we decided to perform the prediction using other models such as Decision Tree, Clustering using K-means.

4. Decision Tree

The model is designed to predict the outcome variable 'Arrest' based on the features. Initially, the categorical variables such as CODE_DEFINED and LarcenyCode are handled using StringIndexer for converting categorical data into numerical indices. OneHotEncoder is utilized to convert the indexed categorical data into a format suitable for model input. VectorAssembler is employed to assemble all the generated features along with additional numerical columns like 'HourStart', 'MinuteStar', 'Duration', 'Day', 'Month', and 'Yea' into a single feature vector named "features". A Pipeline is created to sequentially execute the defined stages of data transformation and model building. Here we used the initial cleaned data and then applied the feature engineering steps and split the training and test date (70%, 30%) respectively. The model's accuracy is assessed using the Multiclass Classification Evaluator, measuring the proportion of correctly predicted instances among the total instances. We obtained the following results.

```
Accuracy = 0.902835
Precision: 0.8871383410452295
Recall: 0.9028348614184846
F1 Score: 0.8912179762224481
```

These metrics collectively indicate that the Decision Tree Classifier performed quite well, achieving a high level of accuracy of 90%, precision - 88%, recall - 90%, and a balanced F1 Score 89%.

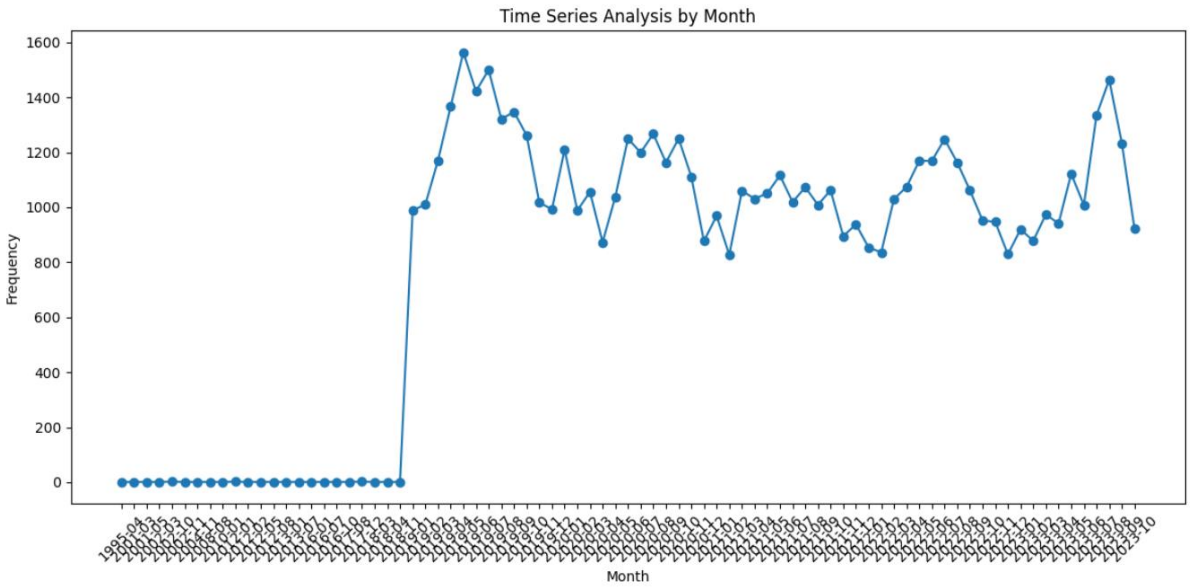
5. K-Mean Clustering

We have utilized the Clustering model on the 'ADDRESS' column in our crime dataset and each address to show which cluster each street name belongs to, aiding in understanding the grouping of similar street names based on the features generated by the HashingTF technique and the K-Means clustering algorithm. Initially, we have created a new column 'streetName' in the DataFrame 'crime' without the leading street numbers using regular expressions, then we tokenized the streetName column. This step splits the column, containing street names without numbers, into individual words. It generates a new column named tokens containing lists of words derived from the streetName column. Then converts the tokenized words into numerical feature vectors (features) using the HashingTF (Hashing Term Frequency) technique. The 'numFeatures' parameter specifies the size of the feature vectors. Initializes a K-Means clustering algorithm. It is set to identify 5 clusters. Following are the results of K-means clustering. The prediction column gives the occurrences of Arrest in that location.

streetName	prediction
BALLANTYNE RD	2
DAWES AV	2
FENTON ST	1
MILDRED AV	2
MIDLAND AV	2
W GENESEE ST	0
CLYDE AV	2
GENESEE PARK DR	2
BRUCE ST	1
ROWLAND ST	1
S SALINA ST	0
SOLAR ST	1
CANNON ST	1
BUTTERNUT ST	1
S GEDDES ST	0
E LAUREL ST	0
SOUTH AV	2
ACKERMAN AV	2
DESTINY USA DR	2
DESTINY USA DR	2

6. Time Series Analysis

Initially, we generated a time series analysis that delved into the occurrences of crimes across each month of the year. This analysis specifically depicted the count of crimes recorded within every individual month.



The visual representation indicates a notable trend in crime occurrences over time. In 2019, the data illustrates a relatively high frequency of crimes, gradually tapering towards the year-end. This pattern suggests a potential decrease in criminal activities during the Christmas holidays or winter season. Furthermore, a discernible peak in crime incidents appears evident during the months of June and July in the ongoing year, 2023.

RESULTS SUMMARY

Model	Precision	Recall	F1-Score	Accuracy
Random Forest	1.0	1.0	1.0	1.0
Logistic Regression	1.0	0.5774	0.7321	0.5774
Decision Tree	0.8942	0.9077	0.8981	0.9077

Random Forest Achieves perfect scores (1.0) for all metrics (accuracy, precision, recall, F1-score). This may imply that the Random Forest model perfectly predicted the outcomes for the given test data.

The Decision Tree model also performed well, with high precision, recall, and F1-score, but not perfect.The Logistic Regression model had significantly lower recall and F1-score, suggesting it didn't perform as well as the other two models in classifying the positive class. The Random Forest model seems to perform the best based on the provided metrics. However, due to the perfect scores, it is advisable to validate the results further to ensure they are reliable and

the model is robust.The Random Forest model seems to perform the best based on the provided metrics. However, due to the perfect scores, it is advisable to validate the results further to ensure they are reliable and the model is robust.

PROBLEMS ENCOUNTERED

At the outset, we encountered numerous challenges during the data cleaning and preprocessing stages. The dataset, sourced from the City of Syracuse website, presented inaccuracies attributed to privacy protection measures for Syracuse residents by the city's department. Consequently, models like Linear Regression couldn't deliver optimal outcomes due to the presence of unreliable numerical data. Moreover, executing operations on such a voluminous dataset within Google Colab posed difficulties, causing some issues while running the models. Despite these hurdles, we were pleasantly surprised by the remarkable performance of certain models, enabling us to extract valuable insights and provide recommendations based on thorough data analysis and our dataset's initial exploration.

CITATIONS

1. https://data.syr.gov/pages/open-data-inventory
2. https://data.syr.gov/datasets/d3c98278e2864a2bbcd00e6e30358856_0/about
3. https://www.arcgis.com/sharing/rest/content/items/d3c98278e2864a2bbcd00e6e30358856/info/metadata/metadata.xml?format=default&output=html