Machine Learning Framework for Crime Prediction: Integrating Socioeconomic Factors and Geo-spatial Analysis (Configuration Manual)

Mary Cindrilla Moreira
x22114386@student.ncirl.ie

Dec-2023

1 Introduction

You have arrived at the Machine Learning Framework for Crime Prediction: Integrating Socioeconomic Factors and Geo-spatial Analysis Configuration Manual. For the effective deployment and operation of the Machine Learning Framework for Crime Prediction: Integrating Socioeconomic Factors and Geo-spatial Analysis system, this paper provides a thorough reference to the setup, parameters, and requirements needed. For a smooth configuration procedure, this handbook offers crucial information for all users, developers, and administrators alike.

2 Purpose

This manual's main goal is to make setting up Project easier by offering detailed instructions, industry best practices, and insights into the different parts that comprise the system. Achieving the intended functionality of the system, guaranteeing security, and maximizing performance all depend on proper setup.

3 Hardware Requirements

The following settings were used for the project's implementation in Local Machine

- **RAM:** 20.0 GB for effective multitasking
- **System Type:** 64-bit OS, x64-based CPU for improved performance
- **Processor:** 1.60 GHz (1.80 GHz turbo) Intel Core i5-8250U for a power-efficiency balance
- **Storage:** 256 GB SSD for dependable and quick storage
- **Operating System:** Windows 11 Home Single Language edition for a user-friendly experience

4 Software Requirements

4.1 Python

Python version of 3.12 is used

4.2 Visual Studio

5 Dataset Specifications

5.1 Dataset 01: "PoliceStationsOfNewYorkCity.csv"

Information for 77 police precincts was gathered from the official NYPD website (https://www.nyc.gov/site/nypd/bureaus/patrol landing.page), according to the dataset. The "Precinct," "Phone," "Address," and "Borough" columns are among those in this dataset.

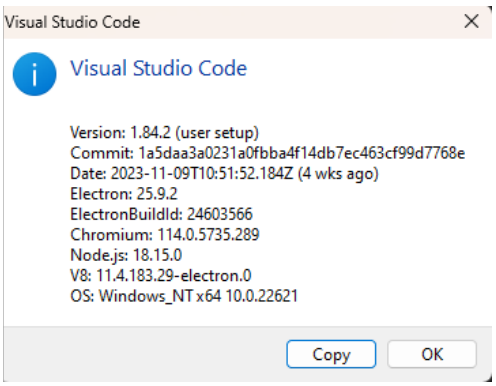

Figure 1: Visual Studio Configuration.

Library	**Purpose**
Pandas (`import pandas as pd`)	Used for data manipulation and analysis.
Matplotlib (`import matplotlib.pyplot as plt`)	Used for data visualization.
Seaborn (`import seaborn as sns`)	Built on top of Matplotlib, used for statistical data visualization.
Geopandas (`import geopandas as gpd`)	Extends Pandas to enable spatial operations and mapping.
Geopy (`from geopy.geocoders import Nominatim`)	Used for geocoding addresses.
Shapely (`from shapely.geometry import Point`)	Provides geometric objects like Point, Polygon, etc., for spatial analysis.
Folium (`import folium`)	Used for interactive maps.

Table 1: Python Libraries for Data Analysis and Visualization

5.1.1 Libraries that are imported

5.1.2 Others

Visualization of maps:

Provide more details on the map visualization procedure, particularly if it involves reliance on outside files (such GeoJSON data). that is attached with this project file "nyc_precincts.geojson" HTML Export:

Provide instructions on how to save and read the HTML file if the map's HTML export is significant. The .html map file was generated.

5.2 Dataset 02: "PopulationAreaWithGDP.csv"

The dataset includes the results of the 2020 census of population for New York City and was obtained from the QuickFacts website of the Census Bureau (https://www.census.gov/quickfacts/fact/table/newyorkcitynewyork/PST045222). It offers information on land area in square miles and kilometers, population density per square mile and per square kilometer, counties, boroughs, and census counts. Additional columns such as "Billions" and GDP in US dollars for 2012 are given, however it is unclear why these are specifically relevant.

Library	Purpose
Pandas (`import pandas as pd`)	Data manipulation and analysis
Matplotlib (`import matplotlib.pyplot as plt`)	Data visualization
Seaborn (`import seaborn as sns`)	Statistical data visualization
Geopandas (`import geopandas as gpd`)	Spatial operations and mapping
Geopy (`from geopy.geocoders import Nominatim`)	Geocoding addresses
Shapely (`from shapely.geometry import Point`)	Geometric objects for spatial analysis
Folium (`import folium`)	Creating interactive maps

Configuration Point	Description
File Paths	Provide correct paths to CSV files for data loading
Data Cleaning	Be aware of missing values and duplicate rows
Geocoding Configuration	Adjust geocoding parameters (user agent, timeout)
Visualization	Customize plots based on preferences
Dependency Installation	Ensure necessary packages are installed
Map Output	Save Folium map as HTML or display in Jupyter
Dataset Description	Briefly describe dataset columns and content
Data Types and Statistics	Print information about data types and statistics
Spatial Analysis	Ensure availability of necessary geographic data files
Customization	Modify plot parameters as needed
External Data Sources	Verify availability of required external files

5.3 Dataset 03: "CrimeNYC.csv"

A thorough record of all occurrences reported to the New York Police Department (NYPD "https://data.cityofnewyork.us/Public-Safety/nypd/pv2jzure"- Primary dataset) is contained in the dataset, which was obtained from the NYPD dataset. An individual complaint number (CMPLNT_NUM) is assigned to each incidence. Critical temporal data is provided by the dataset (CMPLNT_FR_DT and CMPLNT_FR_TM), which includes the time and date of the occurrences' original reporting. CMPLNT_TO_DT and CMPLNT_TO_TM record an incident's end date and time if it can be determined. Consisting of the precinct code, the ADDR_PCT_CD column provides information on the occurrences' geographic location.

Other important characteristics are KY_CD, a numerical number that indicates if the occurrence is a misdemeanor or a felony, and RPT_DT, which indicates the official reporting date. Numerous topics are covered by the dataset, including the premises type (PREM_TYP_DESC), jurisdiction-related information (JURIS_DESC, JURISDICTION_CODE), and location description (LOC_OF_OCCUR_DESC). Spatial analytic capabilities are further enhanced by geographic coordinates (Latitude, Longitude) and the combined Lat_Lon column.

Suspect demographics, including age group, race, and gender (SUSP_GROUP), as well as victim demographics (VIC_AGE_GROUP, VIC_RACE, and VIC_SEX), help to provide a complete picture of law enforcement operations. The complexity of the dataset makes it possible to examine recorded incidents in great depth, which helps develop ideas and plans for improving public safety in New York City.

Libraries Imported	Purpose
pandas	Data manipulation and analysis
matplotlib.pyplot	Creating visualizations in Python
seaborn	Statistical data visualization
plotly.graph_objects	Creating interactive visualizations
chart-studio	Publishing interactive plots online

Table 2: Imported Libraries

Configuration Manual Steps	Details
Dataset Loading	Specify the path and use `pd.read_csv` to load the dataset into a DataFrame
Data Exploration	Check dataset information using `df.info()` and visualize missing values
Missing Data Analysis	Calculate missing data proportions, visualize patterns, use different libraries
Data Cleaning	Optionally clean data based on missing data analysis, create a backup
Visualization	Create visualizations for data distribution and patterns
Library Installation	Include library installation using `!pip install`
Additional Configurations	Specify additional settings and customization instructions
Code Organization	Emphasize code organization, use functions for better readability

Table 3: Configuration Manual Steps

Libraries Used in Data Cleaning	Purpose
pandas	Data manipulation and analysis
matplotlib.pyplot	Creating visualizations in Python
numpy	Numerical operations
dateutil.parser	Parsing date strings into datetime objects
sklearn.preprocessing	Standardizing and normalizing numeric data

Table 4: Libraries Used in Data Cleaning

5.4 Dataset 04: "CompleteDs.csv"

This dataset was collected by web scraping from multiple reliable data sources, and it includes the columns that were specified. MacroTrends (https://www.macrotrends.net/cities/23083/new-york-city/population) and the official NYC Planning historical population report (https://www.nyc.gov/assets/planning/download/pdf/data-maps/nyc-population/historic population/nyc_total_pop_1900-2010.pdf) were the sources of the population data from 1950 to 2019. Working-age group data were obtained from https://fred.stlouisfed.org/series/LFWA64TTUSM647S, the Federal Reserve Economic Data (FRED). The sources of the poverty rates were the HHS Poverty Guidelines (https://aspe.hhs.gov/topics/poverty-economic-mobility/poverty-guidelines/prior-hhs-poverty-guidelines-federal-register-references) and the Wikipedia article on New York City's demographics (https://en.wikipedia.org/wiki/Demographics_of_New_York_City). Income data was obtained from FRED (https://fred.stlouisfed.org/series/NYPCPI), while unemployment rates were gathered from FRED (https://fred.stlouisfed.org/series/NYUR). Crime-related housing price change data was added, and FRED's (https://fred.stlouisfed.org/series/NYPCPI) educational data for New York City was taken into consideration.

The aforementioned dataset is extensive, spanning several decades and containing essential measures that provide a nuanced knowledge of the socio-economic conditions in several New York City boroughs. The data is carefully compiled from reliable sources to assure accuracy and dependability, which makes it a great tool for in-depth analysis and well-informed decision-making. (

Libraries Imported	Purpose
pandas	Data manipulation and analysis
matplotlib.pyplot	Creating visualizations in Python
seaborn	Statistical data visualization
plotly.graph_objects	Creating interactive visualizations
chart-studio	Publishing interactive plots online

Table 5: Imported Libraries

5.5 Dataset 05: "merged_dataFinal.csv"

This dataset is an extensive aggregation of data from CompleteDs.csv and CrimeNYC.csv, two main sources. The first step involved data translation of the CrimeNYC.csv dataset, which contained 11 million records. By extending the dataset's dimensions, namely by averaging crime categories such as "Felony," "Misdemeanor," and "Violation," the dataset was flattened. After that, the flattened dataset was organized by years and Boroughs, which led to a major decrease in the number of records—from 11 million to 350—making it easier to handle. Ultimately, a thorough and integrated summary was produced by merging this compressed dataset with the original CompleteDs.csv file.

Imported Libraries	Purpose
pandas	Data manipulation and analysis
matplotlib.pyplot	Creating visualizations in Python
seaborn	Statistical data visualization
plotly.graph_objects	Creating interactive visualizations
plotly.express	Creating dynamic visualizations

Table 6: Imported Libraries

Configuration Manual Considerations	Details
Dataset Loading	Specify the path and use `pd.read_csv` to load the dataset into a DataFrame
Data Exploration	Check dataset information using `df.info()` and visualize missing values
Missing Data Analysis	Calculate missing data proportions, visualize patterns, use different libraries
Data Cleaning	Optionally clean data based on missing data analysis, create a backup
Visualization	Create visualizations for data distribution and patterns
Custom Visualizations	Specify purpose and interpretation of custom plots
Statistical Analysis	Provide information on methods and reasoning
Dependencies and Environment Setup	Mention necessary libraries and versions, suggest using a virtual environment
Visualization Output	Specify how visualizations will be displayed
Usage Instructions	Provide step-by-step instructions for code execution
Additional Notes	Include any relevant additional information
Time Series Plot	Specify requirements for datetime column
Plotly Express Configuration	Provide details on configuring Plotly Express plots
Interactive Plots	Mention specific features or interactions for interactive plots

Table 7: Configuration Manual Considerations

5.6 Model Building

5.6.1 Simple Linear Regression

Imported Libraries	Purpose
pandas	Data manipulation and handling DataFrames
numpy	Numerical operations and transformations
sklearn.model_selection	Splitting the dataset into training and testing sets
sklearn.linear_model	Implementing the Linear Regression model
sklearn.metrics	Providing metrics for model evaluation
matplotlib.pyplot	Creating visualizations, especially scatter plots
seaborn	Enhancing the aesthetics of visualizations
scipy.stats	Calculating Z-scores
MinMaxScaler, StandardScaler (sklearn.preprocessing)	Feature scaling

Table 8: Imported Libraries

5.6.2 Random Forest

5.6.3 Ordinary Least Square Model

5.6.4 K Nearest Neighbour

6 Code Repository

The complete code and data is available in https://github.com/x22114386/NewYorkCrimeAndSocioEconomicFactors.git

Configurations	Details
Data Splitting	Use `train_test_split` for training and testing sets
Model Selection	Choose Linear Regression, explain suitability
Features	Specify selected features and rationale
Model Training	Train linear regression on the training set
Evaluation	Use MSE, R-squared for model evaluation
Visualization	Emphasize predicted vs. actual values
Feature Engineering	Describe 'InteractionTerm', 'SquaredPopulation', 'LogPopulation'
Outliers	Use Z-scores for outlier identification
Scaling	Explain Min-Max, Standard Scaling on features
Target Loop	Iterate over target variables
Results	Present and interpret evaluation and visualizations

Table 9: Configuration Manual Considerations Simple Linear regression.

7 Appendix On Analysis(Graphs)

7.1 Dataset 01:"PoliceStationsOfNewYorkCity.csv"

The plot displays a bar chart with the bars arranged according to the number of police stations in each borough, illustrating how the stations are distributed throughout the several boroughs. The x-axis labels have been rotated for better readability, and the figure is the right size with clear labels. Plot: The base map shows the NYC precincts

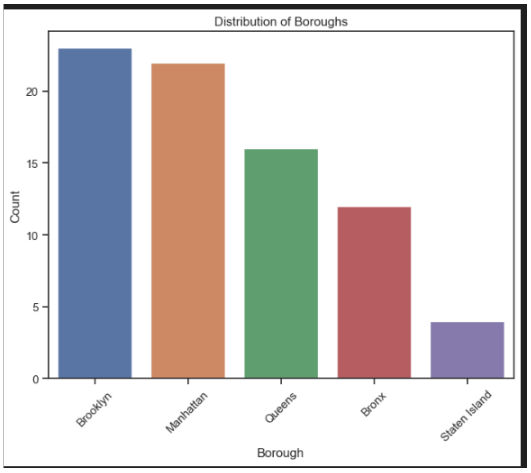

Figure 2: police station distribution.

highlighted in red, with labels for the title, longitude, and latitude clearly visible. Using Geopy and Nominatim, the code geocodes addresses from address_borough_array to extract (latitude, longitude) coordinates for each address; these coordinates are then placed in the coordinates list, which may be used for further analysis or plotting on a map.

7.2 Dataset 02: "PopulationAreaWithGDP.csv"

7.3 Dataset 03: "CrimeNYC.csv"

7.4 Dataset 04: "CompleteDs.csv"

7.5 Dataset 05: "merged_dataFinal.csv"

7.6 Data Modelling and Results

Configuration Manual Considerations	Details
Data Splitting	Split the data into training and testing sets using `train_test_split`. Specify features and target variables.
Model Selection	Choose `RandomForestRegressor` for regression tasks. Adjust parameters like `n_estimators` based on your dataset.
Model Training	Initialize and fit the `RandomForestRegressor` model on the training set.
Model Evaluation	Use mean squared error (`mean_squared_error`) and R-squared (`r2_score`) for evaluation.
Visualization	Plot predicted vs. actual values for each target variable to assess model performance.
Feature Importance	Print and analyze feature importance using `model.feature_importances_`. Optionally, create visualizations.
Code Organization	Emphasize code organization and comments for better readability. Encourage the use of functions or modular code.
Library Versions	Include versions of used libraries, considering potential variations in functionalities.
Usage Instructions	Provide step-by-step instructions on running the code. Specify any configurable parameters.
Additional Considerations	Include any relevant notes or considerations for users.
Visualization Output	Specify how visualizations will be displayed (inline, saved, external tools).

Table 10: Configuration Manual Considerations for Random Forest

Configuration Manual Considerations	Details
Data Splitting	Split the data into training and testing sets using `train_test_split`. Specify features and multiple target variables.
Model Selection	Use Ordinary Least Squares (OLS) regression for each target variable.
Model Training	Build separate OLS models for each target variable. Add a constant term to the independent variables.
Model Evaluation	Print summary statistics for each OLS model. Evaluate using mean squared error (`mean_squared_error`) and R-squared (`r2_score`).
Visualization	Plot predicted vs. actual values for each OLS model.
Library Versions	Include versions of used libraries, considering potential variations in functionalities.
Usage Instructions	Provide step-by-step instructions on running the code. Specify any configurable parameters.
Additional Considerations	Include any relevant notes or considerations for users.
Visualization Output	Specify how visualizations will be displayed (inline, saved, external tools).

Table 11: Configuration Manual Considerations for OLS Models

Configuration Manual Considerations	Details
Data Selection	Select relevant spatial features (`Latitude`, `Longitude`, `Population_density_persons_per_sq_km`, `Poverty_Rate`, `UnEmployment_Rate`) and the target variable (`Bor_Names`).
Label Encoding	Encode borough names (`Bor_Names`) to numeric labels using `LabelEncoder`.
Data Splitting	Split the data into training and testing sets using `train_test_split`.
Model Selection	Choose K-Nearest Neighbors (KNN) classifier with a specified number of neighbors (e.g., 3).
Model Training	Create and train the KNN classifier using `fit` method.
Folium Map Creation	Create a Folium map centered on New York City (`crime_map`).
Marker Clusters	Create MarkerClusters for true and predicted crime locations.
True Crime Locations	Plot true crime locations on the map using green markers.
Predicted Crime Locations	Plot predicted crime locations on the map using red markers.
Display the Map	Display the generated map.

Table 12: Configuration Manual Considerations for Crime Prediction Map

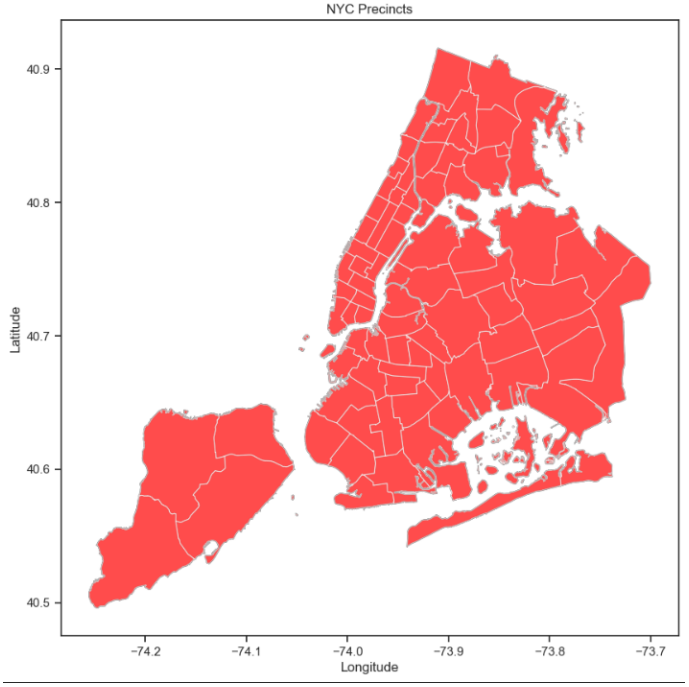

Figure 3: NYC map.

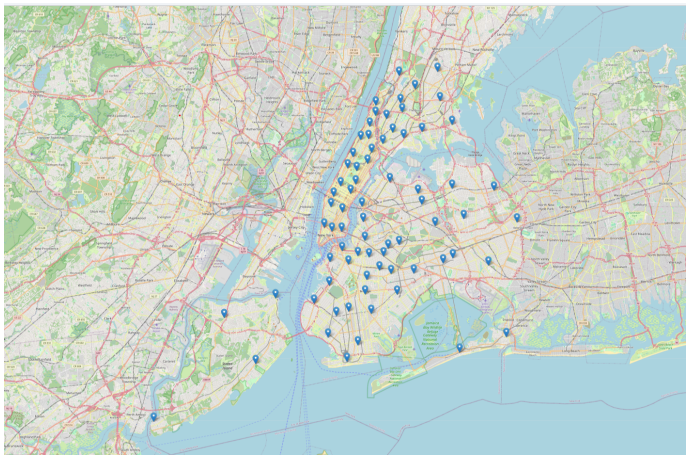

Figure 4: Distribution of Police station in NYC as a HTML page.

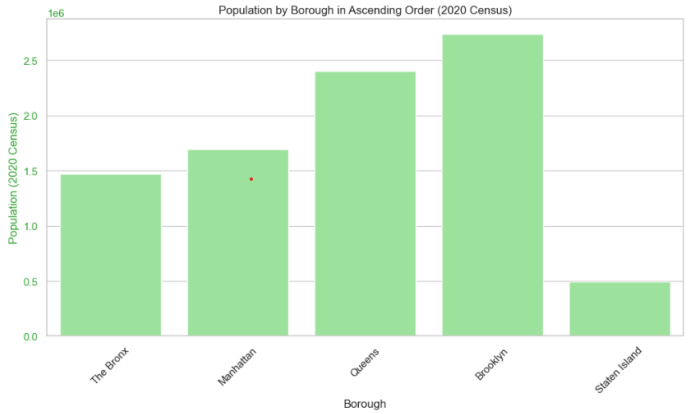

Figure 5: Distribution of population by Boroughs.

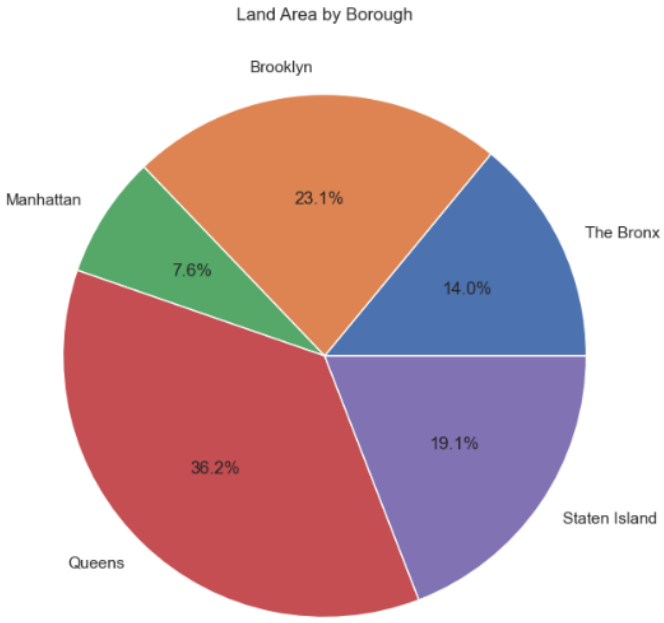

Figure 6: Distribution of Land area by Boroughs.

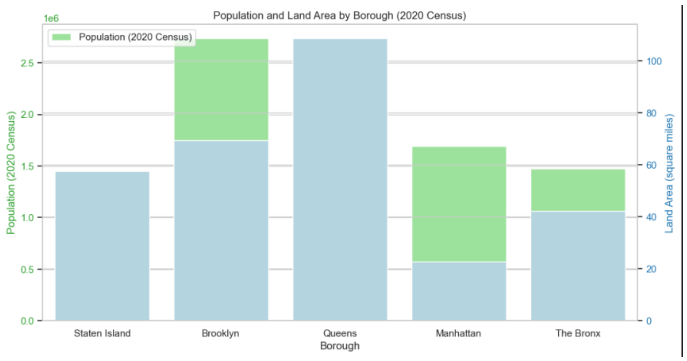

Figure 7: Distribution of population and land area by Boroughs.

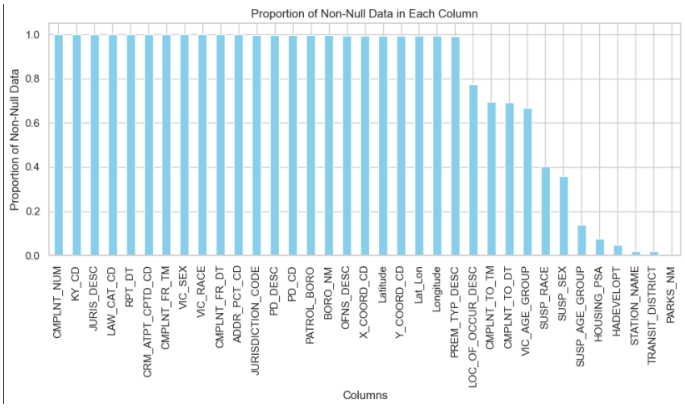

Figure 8: Propotion of Null in each columns.

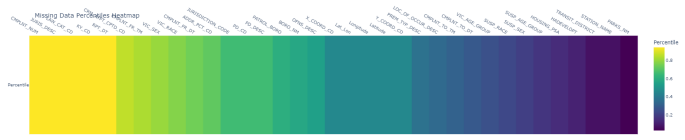

Figure 9: percentage of missing data

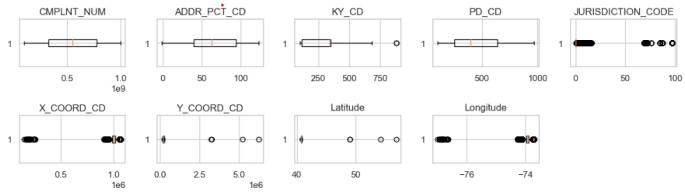

Figure 10: Numeric columns for outliers.

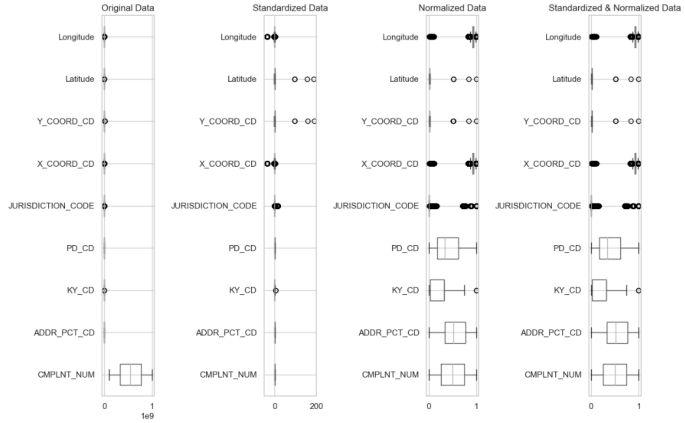

Figure 11: Normalizing and standardizing of data.

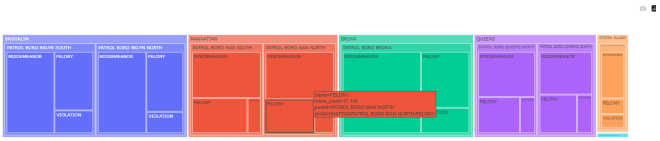

Figure 12: Block size distribution of crime.

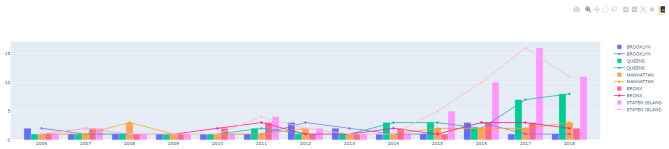

Figure 13: Year wise distribution of crime.

Figure 14: categories of crime.

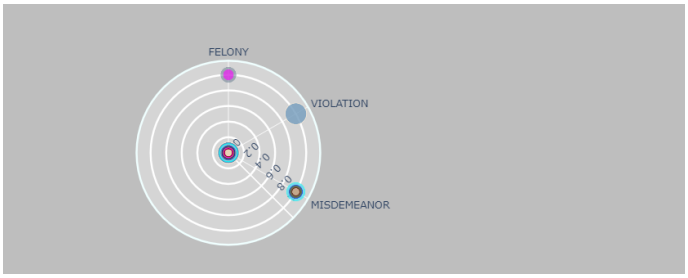

Figure 15: top 10 most crime.

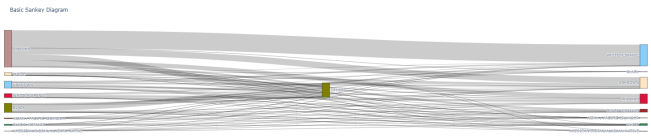

Figure 16: sankey for race.

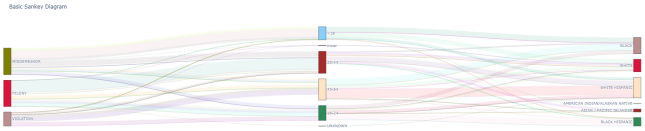

Figure 17: sankey for age and race.

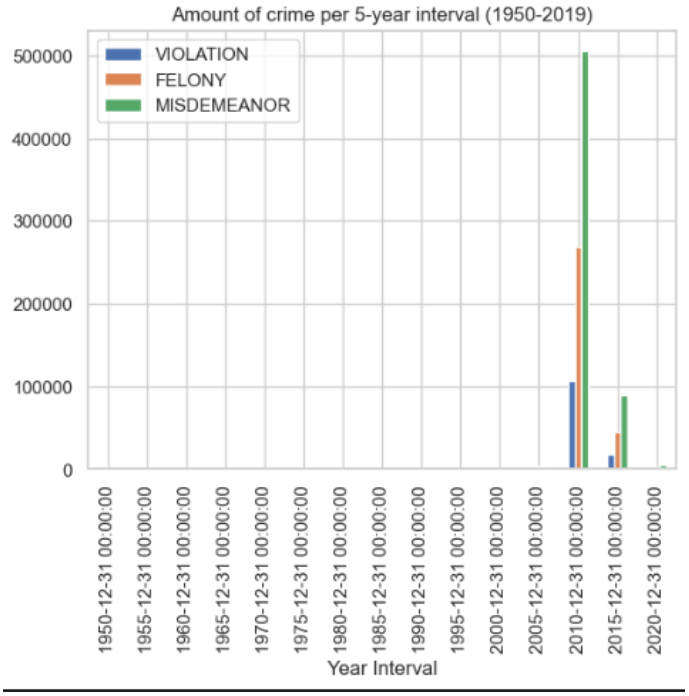

Figure 18: crime interval year.

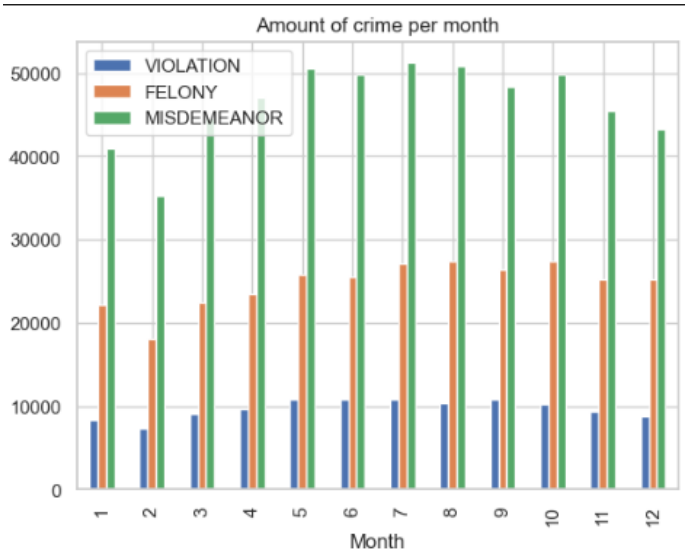

Figure 19: crime interval month.

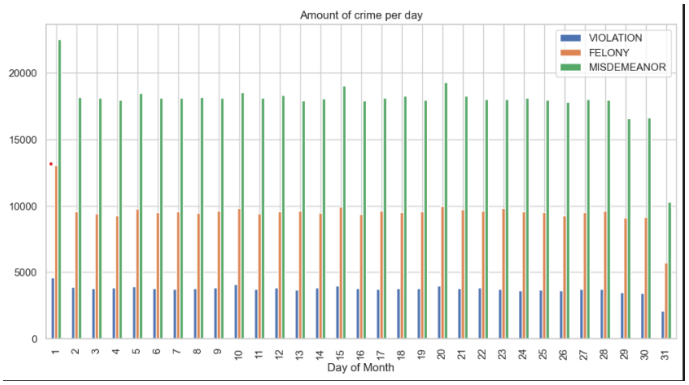

Figure 20: crime interval day.

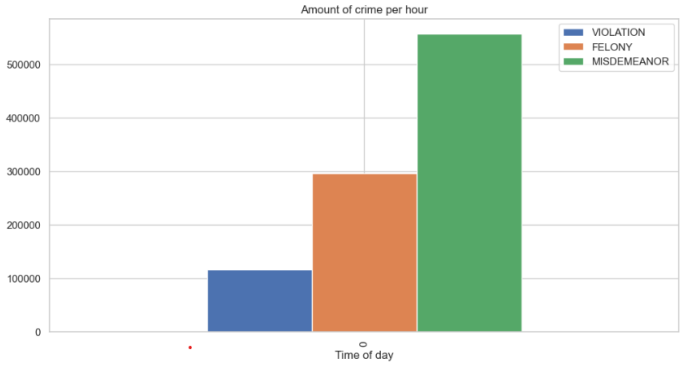

Figure 21: crime interval time.

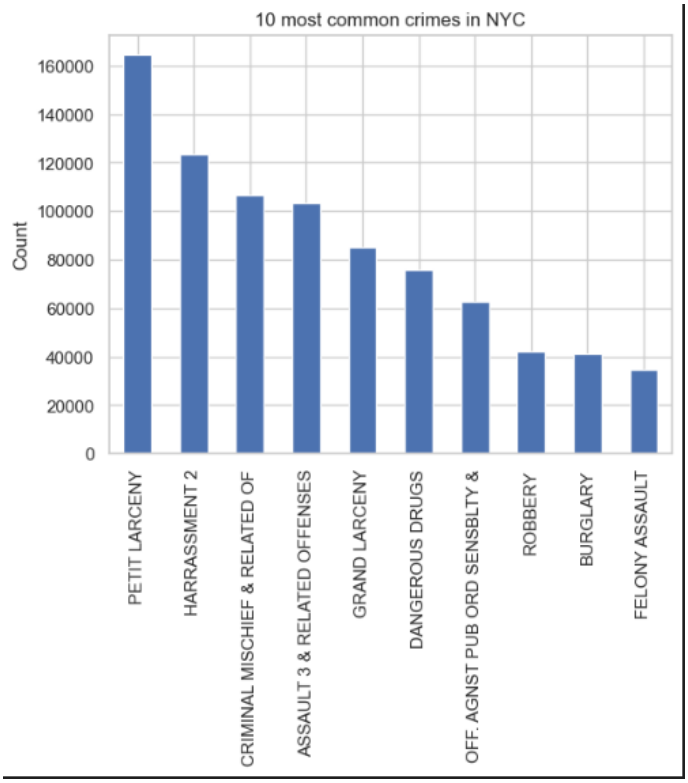

Figure 22: top 10 crimes.

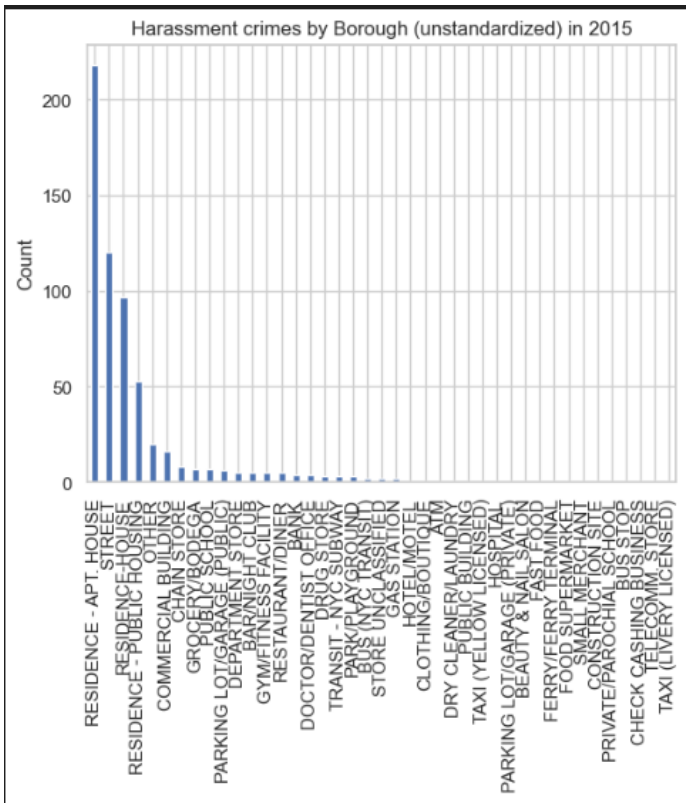

Figure 23: harassment crime based on boroughs

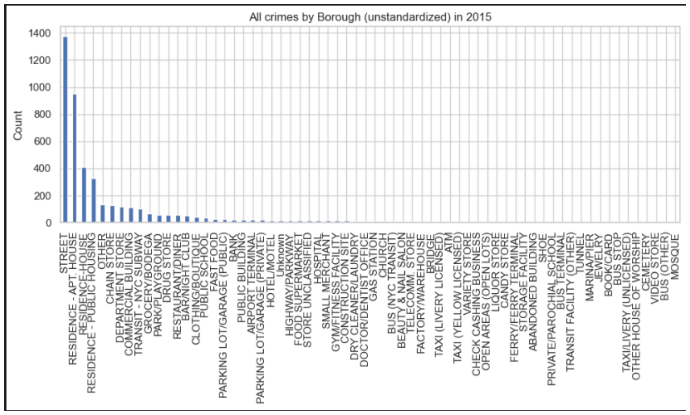

Figure 24: all crimes.

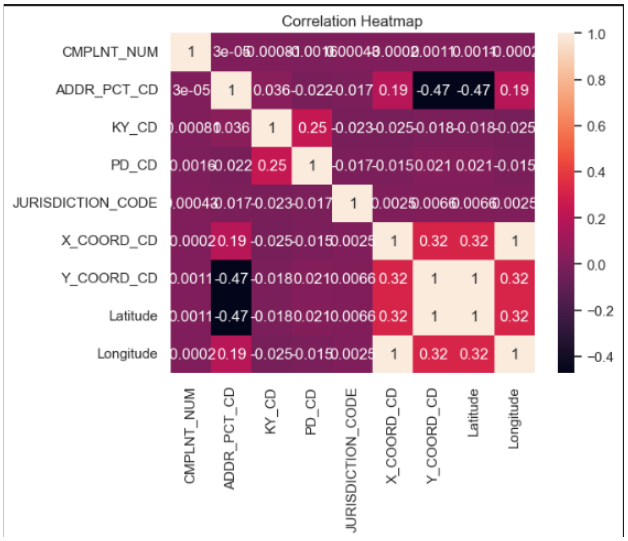

Figure 25: correlation matrix.

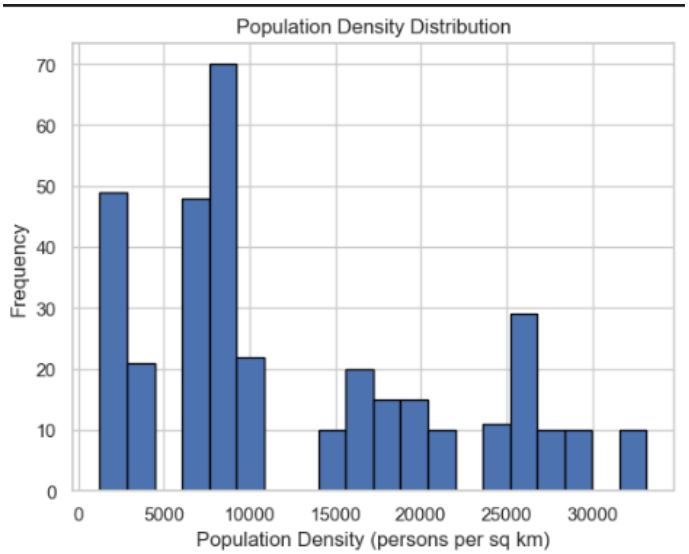

Figure 26: Population density distribution

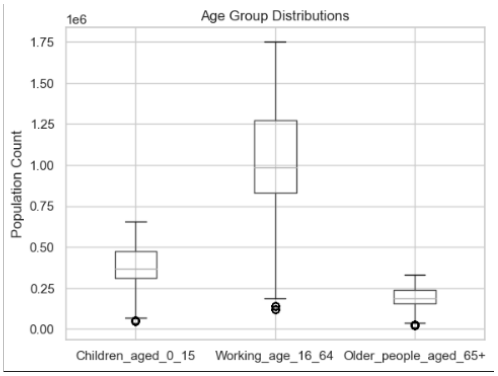

Figure 27: box-plot for age group distribution

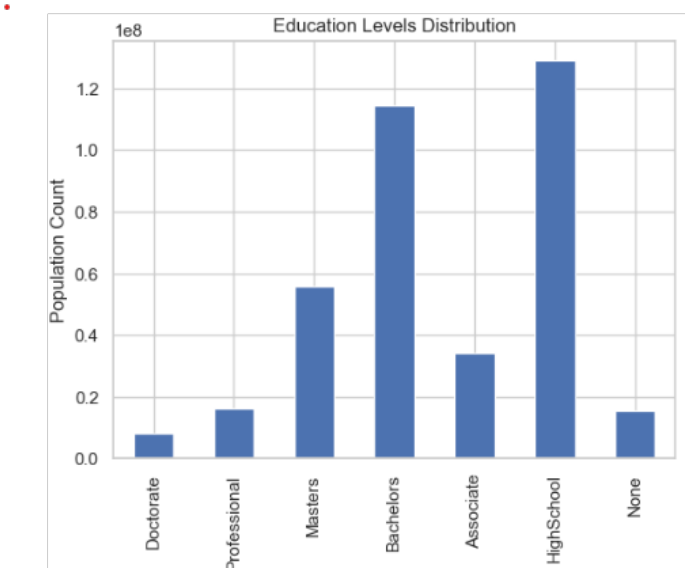

Figure 28: education level distribution.

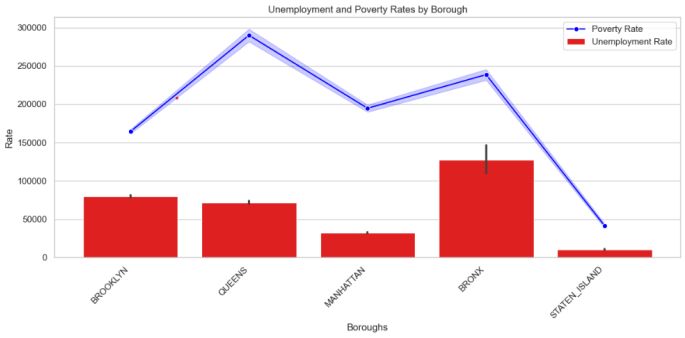

Figure 29: Unemployment and Poverty Rates by Borough

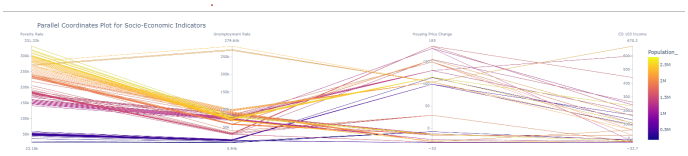

Figure 30: Parallel Coordinates Plot for Socio-Economic Indicators.

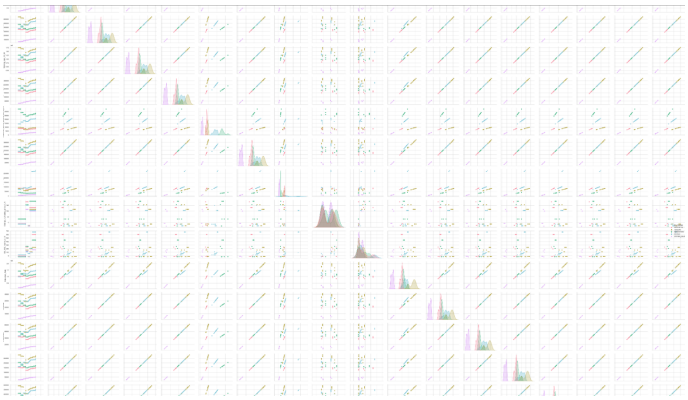

Figure 31: pair plot.

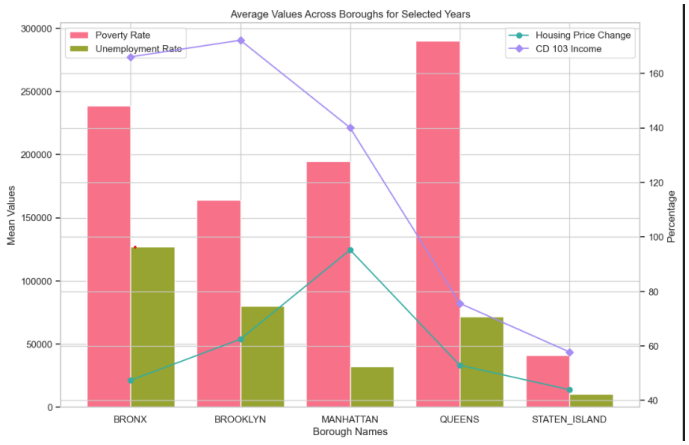

Figure 32: Average Values Across Boroughs for Selected Years

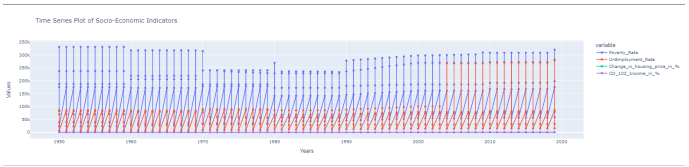

Figure 33: Time Series Plot of Socio-Economic Indicators

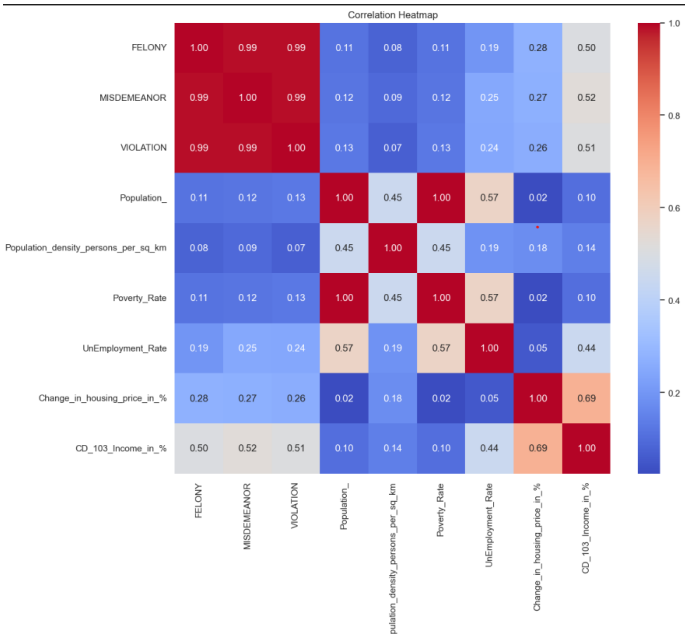

Figure 34: Correlation heat map

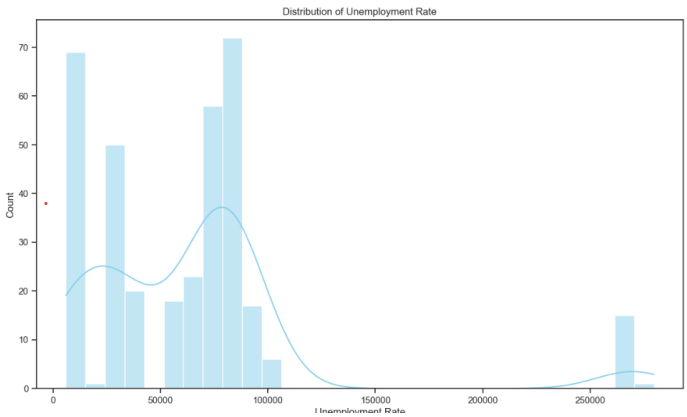

Figure 35: distribution of Unemployment Rate

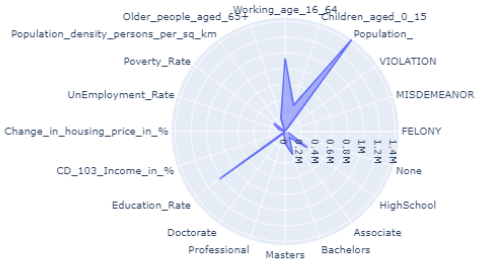

Figure 36: Radar Chart for Crime Rates and Socio-Economic Indicators

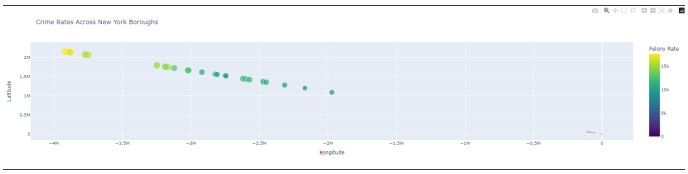

Figure 37: Crime Rates Across New York Boroughs.

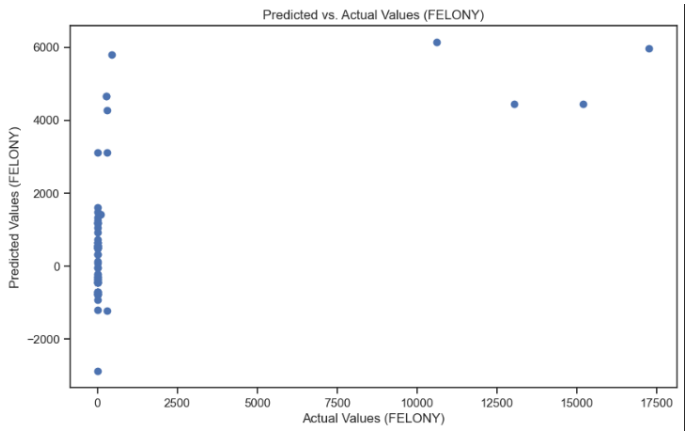

Figure 38: Simple Linear Regression Predicted Vs Actual(Felony).

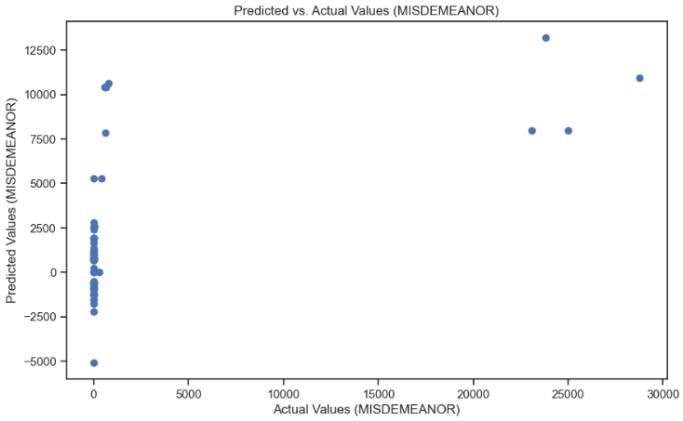

Figure 39: Simple Linear Regression Predicted Vs Actual(Misdemeanor)

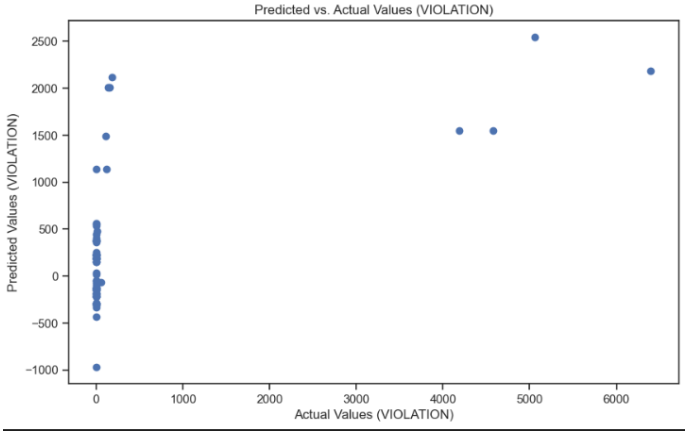

Figure 40: Simple Linear Regression Predicted Vs Actual(Violation).

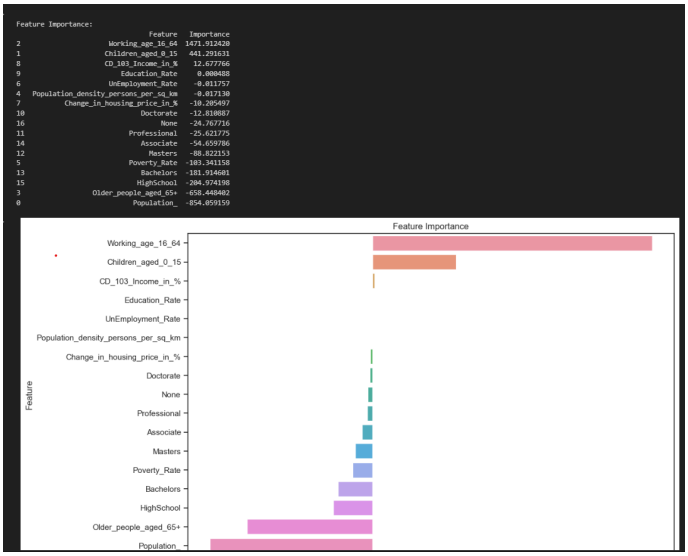

Figure 41: Feature Importance of Simple Linear Regression

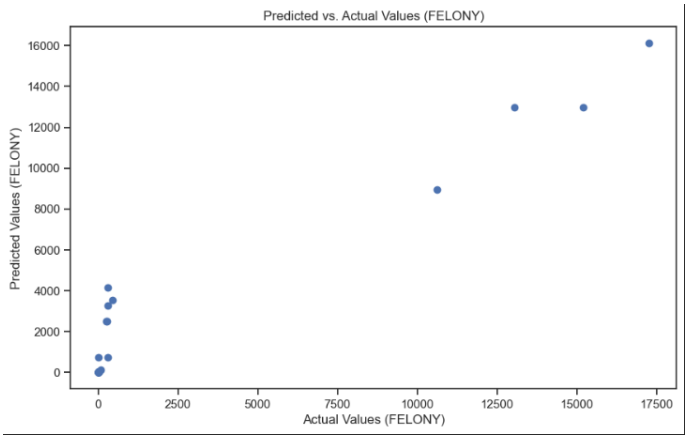

Figure 42: Random Forest Predicted Vs Actual(Felony).

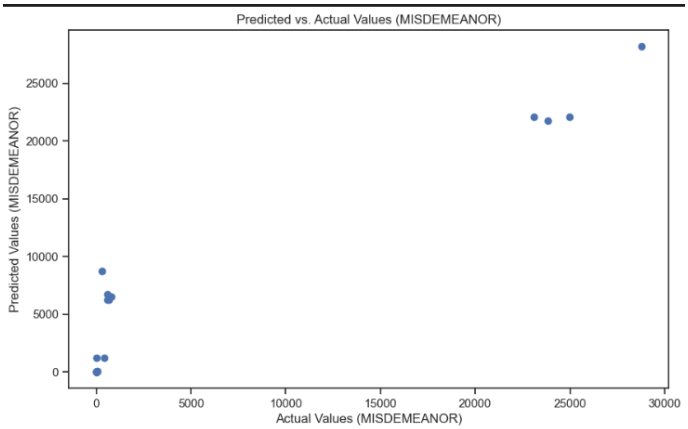

Figure 43: Random Forest Predicted Vs Actual(Misdemeanor)

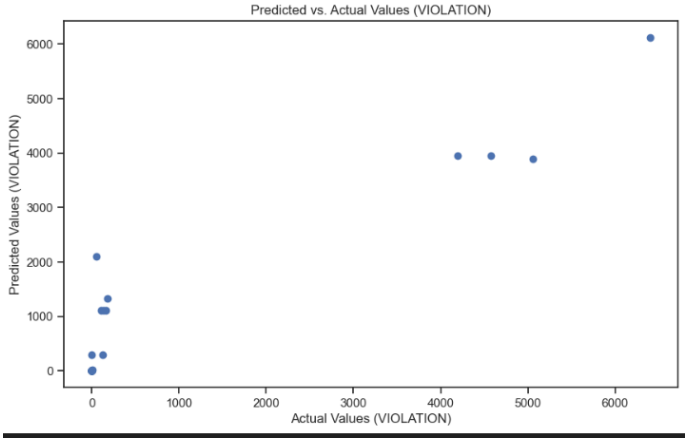

Figure 44: Random Forest Predicted Vs Actual(Violation).

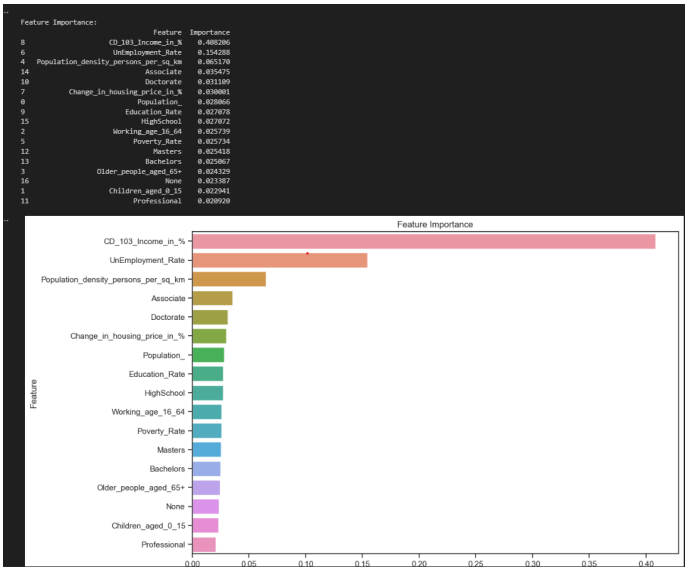

Figure 45: Feature Importance of Random Forest.

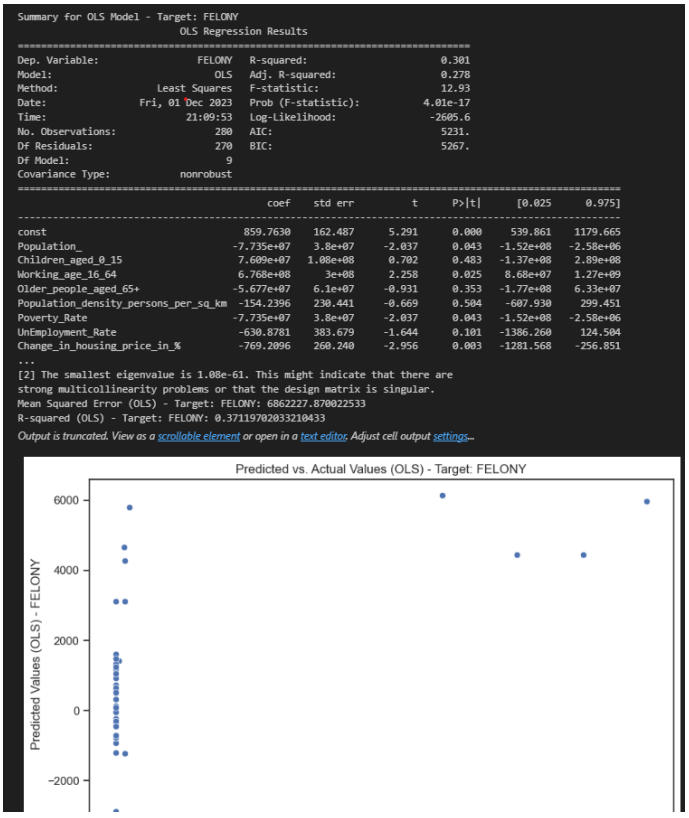

Figure 46: OLS Model For Felony

```
Summary for OLS Model - Target: MISDEMEANOR
                            OLS Regression Results
==============================================================================
Dep. Variable:           MISDEMEANOR   R-squared:                       0.319
Model:                           OLS   Adj. R-squared:                  0.296
Method:                Least Squares   F-statistic:                     14.04
Date:               Fri, 01 Dec 2023   Prob (F-statistic):           1.56e-18
Time:                       21:09:53   Log-Likelihood:                -2776.2
No. Observations:                280   AIC:                             5572.
Df Residuals:                    270   BIC:                             5609.
Df Model:                          9
Covariance Type:           nonrobust
================================================================================================
                                   coef    std err          t      P>|t|      [0.025      0.975]
------------------------------------------------------------------------------------------------
const                         1608.3615    298.817      5.382      0.000    1020.053    2196.670
Population_                  -1.481e+08   6.98e+07     -2.120      0.035   -2.86e+08   -1.06e+07
Children_aged_0_15            1.873e+08   1.99e+08      0.940      0.348   -2.05e+08    5.8e+08
Working_age_16_64             1.239e+09   5.51e+08      2.248      0.025    1.54e+08   2.32e+09
Older_people_aged_65+        -9.36e+07   1.12e+08     -0.834      0.405   -3.14e+08   1.27e+08
Population_density_persons_per_sq_km   -10.4579    423.787     -0.025      0.980    -844.806    823.890
Poverty_Rate                 -1.481e+08   6.98e+07     -2.120      0.035   -2.86e+08   -1.06e+07
UnEmployment_Rate              -409.6775    705.594     -0.581      0.562   -1798.843    979.488
Change_in_housing_price_in_%  -1421.2439    478.588     -2.970      0.003   -2363.483    -479.005
...
[2] The smallest eigenvalue is 1.08e-61. This might indicate that there are
strong multicollinearity problems or that the design matrix is singular.
Mean Squared Error (OLS) - Target: MISDEMEANOR: 20968637.08980299
R-squared (OLS) - Target: MISDEMEANOR: 0.38847664560608086
Output is truncated. View as a scrollable element or open in a text editor. Adjust cell output settings...
```

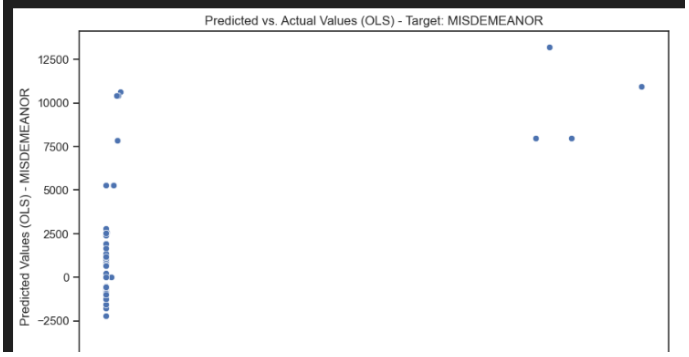

Figure 47: OLS Model For Misdemeanor

```
Summary for OLS Model - Target: VIOLATION
                            OLS Regression Results
==============================================================================
Dep. Variable:              VIOLATION   R-squared:                       0.296
Model:                           OLS   Adj. R-squared:                  0.272
Method:                Least Squares   F-statistic:                     12.60
Date:               Fri, 01 Dec 2023   Prob (F-statistic):           1.07e-16
Time:                       21:09:53   Log-Likelihood:                -2336.8
No. Observations:                280   AIC:                             4694.
Df Residuals:                    270   BIC:                             4730.
Df Model:                          9
Covariance Type:           nonrobust
================================================================================================
                                   coef    std err          t      P>|t|      [0.025      0.975]
------------------------------------------------------------------------------------------------
const                          327.3912     62.223      5.262      0.000     204.888     449.894
Population_                  -2.832e+07   1.45e+07     -1.947      0.053   -5.69e+07    3.18e+05
Children_aged_0_15            2.874e+07   4.15e+07      0.693      0.489    -5.3e+07     1.1e+08
Working_age_16_64             2.467e+08   1.15e+08      2.150      0.032    2.08e+07    4.73e+08
Older_people_aged_65+        -2.063e+07   2.34e+07     -0.883      0.378   -6.66e+07    2.54e+07
Population_density_persons_per_sq_km   -49.9813     88.245     -0.566      0.572    -223.717     123.754
Poverty_Rate                 -2.832e+07   1.45e+07     -1.947      0.053   -5.69e+07    3.18e+05
UnEmployment_Rate              -109.6167    146.925     -0.746      0.456    -398.882     179.648
Change_in_housing_price_in_%   -270.0117     99.656     -2.709      0.007    -466.213     -73.810
...
[2] The smallest eigenvalue is 1.08e-61. This might indicate that there are
strong multicollinearity problems or that the design matrix is singular.
Mean Squared Error (OLS) - Target: VIOLATION: 861849.1482106801
R-squared (OLS) - Target: VIOLATION: 0.38982287754711276
Output is truncated. View as a scrollable element or open in a text editor. Adjust cell output settings...
```

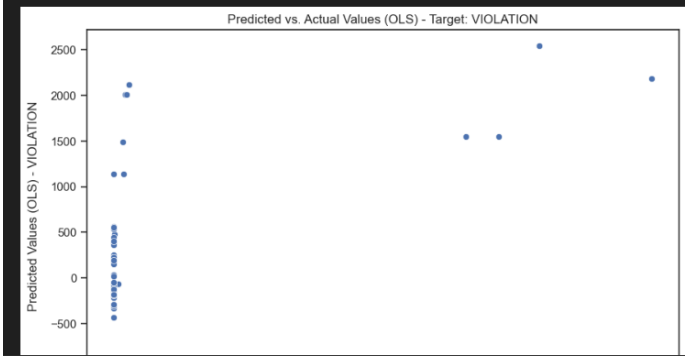

Figure 48: OLS Model For Violation.

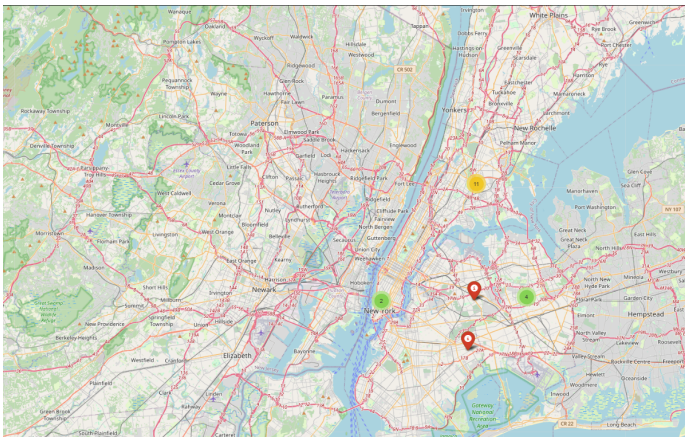

Figure 49: Hotspot using KNN