Montgomery County Crime Analysis

Practical Report (Coursework)

CI7340 Applied Data Programming

Kingston University

Nazanin Zakizadeh K2144935

Sayda Rabia Altaf K2164077

Jeevan Mohan Pawar K2242210

Kantheti Dhana Shravya K2247431

Table of Contents

Abstract— **For each city's police department collects a significant quantity of data every year and maintains track of a wide range of crimes that take place in different places. The National Incident-Based Reporting System (NIBRS) has provided the dataset used in this research study. In this practical project, preliminary data analysis, exploratory data analysis (EDA), and data visualisation are applied to analyses of the dataset. We can learn some fascinating facts and patterns about crime in Montgomery County (Maryland, USA) from this report. These findings will aid in improving law enforcement's understanding of criminal issues and will give them new information that will assist them in better monitoring activity, predicting the risk of accidents, allocating resources efficiently, and making smarter decisions.**

I. Introduction

For this practical project, it is necessary to analyse a dataset of crimes reported in the Montgomery County of Maryland State, USA, to produce information that would be helpful to the government. The National Incident-Based Reporting System (NIBRS) provided this information, which is accurate from 2016 through 2022. [1]

In the previous report, we defined some research questions, which are going to be replied to in this practical work. In this section, the initial data quality assessment will be put into practice using a variety of methods. Then, we'll implement several exploratory data analysis (EDA) methods, to assess the quality of the data, and prepare the data for visualisation. Then we will answer each question by using visualisation methods and libraries.

The authority and the police can coordinate efforts and make choices in the battle against crime with the aid of the hidden insights and patterns identified in this study. Additionally, the study will highlight the weaknesses in the current dataset's quality and offer solutions for how to enhance it. The overall crime rate can be decreased by subsequently using a similar strategy in other areas around the world.

II. Preliminary Data Analysis

The purpose of this section is to examine the raw data, before any changes are made, to determine what kind of Exploratory Data Analysis (EDA) is needed for each variable. By using the info () method, we conclude that there are some null values in multiple numbers of variables, required to be handled in the EDA section. Figure 1 presents the related information. We can see that in 306094 records, we have a considerable number of null values in the Dispatch Date / Time, Crime Name1/2/3, City, Block Address, Zip Code, Sector, Beat, PRA, Address Number, Street Prefix, Street Suffix, Street Type, and End_Date_Time. This method also gives us some information about each variable data type.

```
RangeIndex: 306094 entries, 0 to 306093
Data columns (total 30 columns):
 #   Column                 Non-Null Count   Dtype
---  ------                 --------------   -----
 0   Incident ID            306094 non-null  int64
 1   Offence Code           306094 non-null  object
 2   CR Number              306094 non-null  int64
 3   Dispatch Date / Time   257065 non-null  object
 4   NIBRS Code             306094 non-null  object
 5   Victims                306094 non-null  int64
 6   Crime Name1            305822 non-null  object
 7   Crime Name2            305822 non-null  object
 8   Crime Name3            305822 non-null  object
 9   Police District Name   306000 non-null  object
 10  Block Address          279888 non-null  object
 11  City                   304818 non-null  object
 12  State                  306094 non-null  object
 13  Zip Code               302915 non-null  float64
 14  Agency                 306094 non-null  object
 15  Place                  306094 non-null  object
 16  Sector                 304564 non-null  object
 17  Beat                   304564 non-null  object
 18  PRA                    305855 non-null  object
 19  Address Number         279985 non-null  float64
 20  Street Prefix          13631 non-null   object
 21  Street Name            306093 non-null  object
 22  Street Suffix          5432 non-null    object
 23  Street Type            305755 non-null  object
 24  Start_Date_Time        306094 non-null  object
 25  End_Date_Time          144436 non-null  object
 26  Latitude               306094 non-null  float64
 27  Longitude              306094 non-null  float64
 28  Police District Number 306094 non-null  object
 29  Location               306094 non-null  object
dtypes: float64(4), int64(3), object(23)
memory usage: 70.1+ MB
```

Figure 1. Detailed information of the data frame using df.info ()

Moreover, isna (). sum () method gives us the exact number of null values for each variable, as it is presented in figure 2.

```
Incident ID                  0
Offence Code                 0
CR Number                    0
Dispatch Date / Time     49029
NIBRS Code                   0
Victims                      0
Crime Name1                272
Crime Name2                272
Crime Name3                272
Police District Name        94
Block Address            26206
City                      1276
State                        0
Zip Code                  3179
Agency                       0
Place                        0
Sector                    1530
Beat                      1530
PRA                        239
Address Number           26109
Street Prefix           292463
Street Name                  1
Street Suffix           300662
Street Type                339
Start_Date_Time              0
End_Date_Time           161658
Latitude                     0
Longitude                    0
Police District Number       0
Location                     0
dtype: int64
```

Figure 2. Details of the null values of the columns

To have an initial assessment of the raw dataset, we also use duplicated () method, to make sure there is no duplicated record in the dataset. As the result is presented in figure 3, there are no duplicated records in the dataset.

```
df[df.duplicated()]  # no duplicated record
```

Incident ID	Offence Code	CR Number	Dispatch Date / Time	NIBRS Code	Victims	Crime Name1	Crime Name2	Crime Name3	Police District Name	...	Street Prefix	Street Name	Street Suffix

0 rows × 30 columns

Figure 3. Duplicates in the data frame

A function also is defined in this project to detect the outliers in each numerical data, this function is used for victim, latitude, as well as longitude variables. Some outliers are detected in all three variables. We have kept the outliers in the victim variable since they represent important information in this dataset; however, the outliers in the latitude and longitude variables will be handled by the EDA section.

III. EXPLORATORY DATA ANALYSIS

Data Cleaning

A. Crime Names

According to the description of the dataset, NIBRS Codes have Crime Names associated with them. After analysing the null values of Crime Names, all the null values are divided among 3 NIBRS Codes. Based on official documentation and the information given, each NIBRS Code has a description of the crime name and the crime against details, which is, 26G – Hacking/Computer Invasion - Crime Against Property

35A – Drug/Narcotic Violations - Crime Against Society

90Z – All Other Offences – Crime Against Property/Society/Person

The decision is to replace Crime Name null values using these NIBRS Codes. In the code, the condition is defined based on NIBRS Code and if the crime name is equal to the given code, we use the fillna() method to replace the NaN values according to the UCS documentation.

B. ***Beats/Sectors***

When applied groupby () on beats, sectors, and district names, the output shows that all the null values in these columns are grouped at two district names. From the description of the dataset, it is understood that the sector is a subset of the district and beat is a subset of the sector. There are 10 districts in the given dataset and when grouped by beats and sectors, only beats and sectors corresponding to the TPPD district and districts that belong to the 'Other' category have null values for these columns. Based on this information, the decision is to replace the beat and sector of other districts with others as well. The beats/Sectors that are under TPPD can be replaced with 'Other.' This can be done using the same logic that is used to deal with null values of Crime Names. A condition that checks if the district name of a given row matches either TPPD or Other and the null values of Beat and Sector are replaced using fillna () function.

C. ***Time-related Features***

In this dataset, we have three features in the type of datetime64, The End_Date_Time, and Dispatch Date / Time both have a considerable number of null values, 161658 and 49029, respectively. Regarding the End_Date_Time, we drop out the column, as we are not going to use it, based on our problem statement and the research questions. However; Dispatch Date / Time, with null values for 16% of the data, can give us interesting information regarding the response time.
As a result, we create a separate data frame, with 257065 records, in which the records with null values have been removed by the dropout method. So, we can use this data frame, just for the research question regarding the response time.
In terms of EDA, methods, for the date and time-related features, a function has been defined, to apply the following tasks:

- Put all of the date/time values in the same format of "%m/%d/%Y %H: %M:%S %p"
- Separate the day part from the time part and put them in the newly created columns, with defined names
- Separate the year/month/day from the day column and put them in three newly defined column

In addition, as we are going to use the weekdays, of the Start_date_time, for one of our research questions, the dt.day_name () method is used to create the weekdays from the start date column.

D. ***Police District Name***

For the 'Police District Name' feature, there are 94 null values. To fix these, we investigated the 'Police District Number' and see what is the 'Police District Number' for those missing 'Police District Name'.

We found out that those missing police district names have the following police district numbers:
8.0D,6.0D, 5.0D, 4.0D, 3.0D, and 1.0D
We then filtered those records that have police district names for these police district numbers. We found out that there exists a one-to-one mapping as follows:

‘8D': 'CITY OF TAKOMA PARK',
'6D': 'MONTGOMERY VILLAGE',
'5D': 'GERMANTOWN',
'4D': 'WHEATON',
'3D': 'SILVER SPRING',
'1D': 'ROCKVILLE'

Now, all we must do is replace those invalid police district names by replacing them using the above dictionary.
We take one record, check the police district number, and then replace the null police district name with the respective name from the dictionary accomplish this we applied value_counts () and isna() to count the null values.

E. ***Police District Number***

While replacing the police district names, we figured out that all those records that have null police district names have some input error for police district numbers as well as the numbers for those records are 8.0D,6.0D, 5.0D, 4.0D, 3.0D, 1.0D. If we group police district numbers, we will see that the distribution is something like the following:

```
3D        20066
6D        19344
4D        19200
1D        16658
2D        16034
5D        13768
8D         1590
TPPD       1204
OTHER       212
8.0D         81
3.0D          6
6.0D          2
5.0D          2
1.0D          1
4.0D          1
```

It seems like while data entry for the police district number (in cases where the police district name is missing or null) float or the double value was entered by concatenating it with ‘D’ while in normal cases, an integer value was entered concatenating it with ‘D’ thus those records that have police district names as null, also have police district number with an additional ‘.0’.
We also double-checked this by filtering the city feature for both police district numbers (with and without ‘.0’) and found out that both numbers have the exact same mapping with regard to the other features as well. Thus, we concluded that this is some input error and to fix that we will simply replace those numbers with ‘.0’ with numbers that don't have ‘.0’. We did this using replace function:

df['Police District Number'].replace(to_replace=number, value=replacement, inplace=True)

F. ***Street Type***

We have 339 null values for street type. To fix them, first, we will look at the block address as the block address also has street type as a substring. If for any null street type, we find a block address that isn't empty then we will extract the street type from there and replace the null street type with the extracted street type. For those records that have both street type and block address as null, we will remove those records.

After applying the above strategy, we found out that for every null street type, the block address is also null:
df['Street Type'].isna().sum() == df[df['Street Type'].isna()]['Block Address'].isna().sum() => True
Thus, we will simply remove those records as they are only 339 and not a big chunk.

G. ***Block Address, Street Prefix, Street Suffix, Street Number***

Following is the number of missing values for these features:

Block Address	26206
Address Number	26109
Street Prefix	292463
Street Suffix	300662

Street prefix and street suffix are important features as they don't provide a lot of information. Also, missing data is too great to just remove those records. Thus, we simply drop these features.
To fix null address numbers we can investigate the block address to see if we can extract that from the block address:
df['Address Number'].isna().sum() == df[df['Address Number''].isna()]['Block Address'].isna().sum() => True
Every record that has a null address number, has a null block address, thus we can't extract the address number from the block address. As we have a street name, we don't need an address number, we will simply drop this feature.
We will also drop block address because it has a lot of null values, and we have other features like address name and street type that we can use for our research question instead of using block address which is a combination of address number, address name, and street type.

H. ***City & States***

The provided dataset with reported crimes from Montgomery County in the state of Maryland, USA contains the City and State information. However, after observing the count of different States in the dataset we get the following:

```
MD    305874
16       185
DC        22
VA         4
ME         2
MS         2
```

```
ND          1
15          1
0           1
MC          1
17          1
```

After observing the values, it clearly seems the State column has FIPS instead of the standard Abbreviation of US State, as well as some typos, such as ME, MS, ND, and MC. There is also one record with State as ‘0’ which is not either a valid FIPS or Abbreviation for any of the US states.

In order to fix such records in the dataset, we can utilise the geographical information from the Latitude and Longitude. It can be utilised to get the City and State, and fix the incorrections in the dataset.

I. ***Latitude and Longitude***

Looking at the missing values in the dataset, the Latitude, and Longitude columns seem to have no missing values. However, there are around 2% of the records in the dataset contain values (0, 0), which clearly signifies missing values. These can be fixed by filling up the Latitude and Longitude from the address information available in the form of Street Name, City, and State. These provide scope for filling in the missing Latitude and Longitude values for such records.

We first use a combination of all three Street Name, City, and State to retrieve the location of the address in terms of Latitude and Longitude. In case, the attempt to identify the location fails, we can use the City and State to fill in the values for Latitude and Longitude. After fixing all these records, the number of records fixed for City and State information utilising the location improves further.

The remaining number of records whose locations can’t be found can be safely ignored as the number would be quite low compared to the total number of records in the dataset. [4]

Feature’s Normality Distribution

We can generate a frequency histogram to verify if our numerical data is normally distributed or not. This will guide future statistical research and let us know if the dataset has a normal distribution. Our numerical values for this dataset are as follows:

- 'Victims'
- 'start_year'
- 'start_month'
- 'start_day'
- 'Dispatch_year'
- 'Dispatch_month'
- 'Dispatch_day'
- 'Latitude'
- 'Longitude'

To do so, we use histplot from seaborn, figure 4, as well as KDE parameter, and the blue line shows an estimate of the data distribution, this is what (KDE = True) produces. [2]

A histogram, a traditional visual representation tool, counts the number of data that fit into discrete bins to show the distribution of one or more variables. This function can add a smooth curve created using a kernel density estimate, and normalise the statistic computed within each bin to estimate frequency, density, or probability mass.[3]

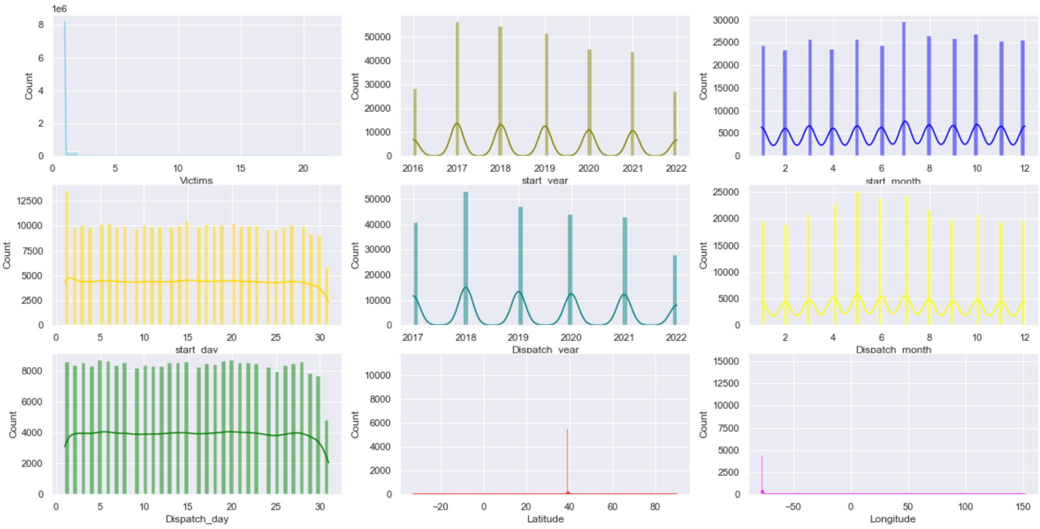

Figure 4. Normality of numerical data in the data frame

If we take a closer look at the start year, start month, dispatch year, and dispatch month looks a bit bell-shaped. Despite the distribution's moderate normality, we cannot categorise it as a normal distribution because of its unevenness.

Data Correlation

In this dataset, most variables are categorical data or numerical variables with no mathematical meaning (such as NIBRS code). It is not feasible to create a useful correlation map for categorical variables. As a result, we will develop a unique connection that is focused only on the numerical values, figure 5. But as it is clear in the heatmap of numerical values correlation, there is not any meaningful relationship between the features, which, for this dataset, makes sense. The only correlation in this dataset is between latitude and longitude, while the high correlation between two variables is beneficial if we can use one variable rather than both. Considering this project requires both latitude and longitude, this correlation map is useless.

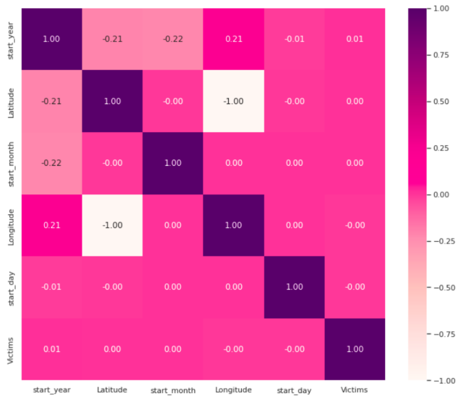

Figure 5. Correlation of numerical data in the data frame

IV. Data Visualisation

The results of the research questions are presented in this chapter using appropriate plots identified in the research report, and the previous coursework.

For research Q1: When do criminals often operate? Which months and days are the busiest? Are there any patterns for these crimes on specific days of the month? Is there a certain month that has the highest or the least criminal activity?

Firstly, we created years, months, and days features by splitting start_date_time column. We then got the count for the month and day columns by using value_counts ().
For months, we have used a pie chart, figure 6. This shows what percentage of crimes are in each month. For months, we have used a pie chart. This shows what percentage of crimes are in each month. This shows us the most active month is May which has criminal activities of around 9.7% and the least active month is February where around 7.3% of total crimes were committed.

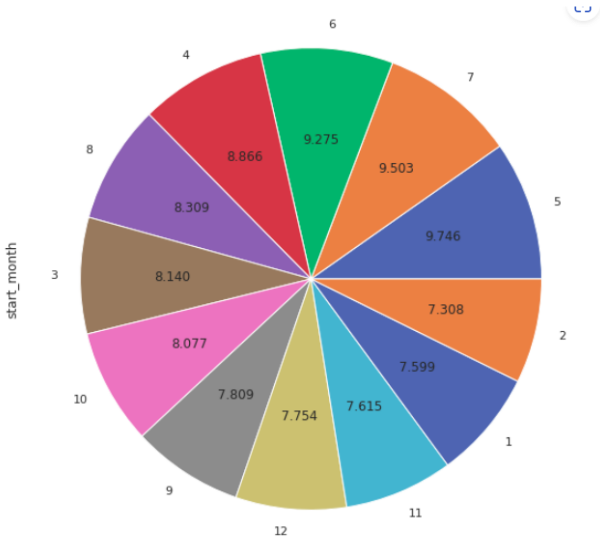

Figure 6. Piechart showing month-wise criminal activities

For days, we used a bar chart, figure 7, which shows that the 1st day of the month has the highest number of crimes and the 31st has the least committed crimes maybe because only 5 months have the 31st.

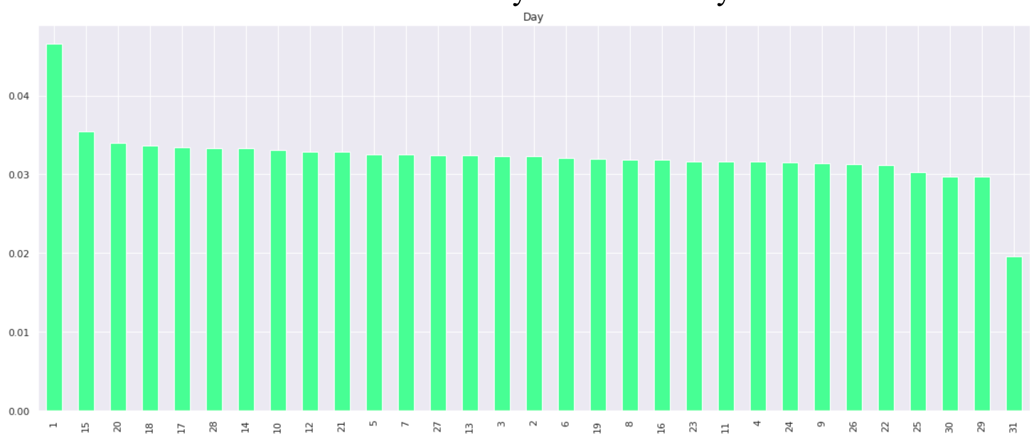

Figure 7. Bar Chart showing day-wise criminal activities

For research Q2: Which crime type has the most affected victims?

To answer this question, we have first found out the total count (in percentages) for each crime type using the count_values function and using normalize=True to normalize the data and show the percentage instead of the total number.

We then plotted the results using the following plots:

Crime Name1: To plot crime name1 we used a pie chart, as it has only 6 categories, which is easily be visualized by a pie chart, figure 8.

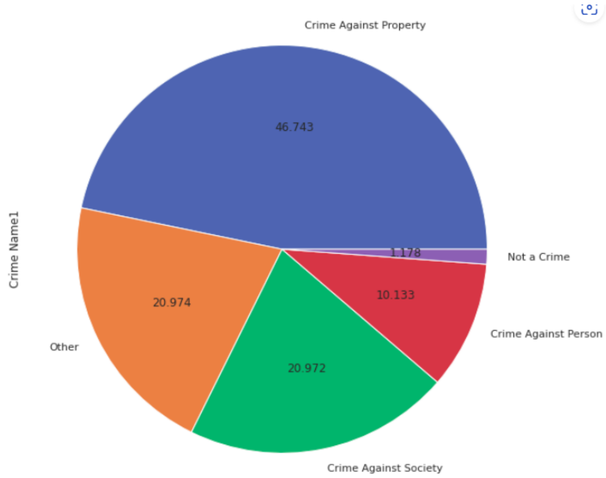

Figure 8. Crime Name1 crime count distribution

It is shown in the chart that 'Crime Against Property' is the most committed crime covering almost 46.8% area of the chart.The least number of crimes committed are those that are not even considered crimes. in this case 'Not a Crime' is around 0.001%

Crime Name2: Crime name2 is plotted using a bar chart, figure 9.

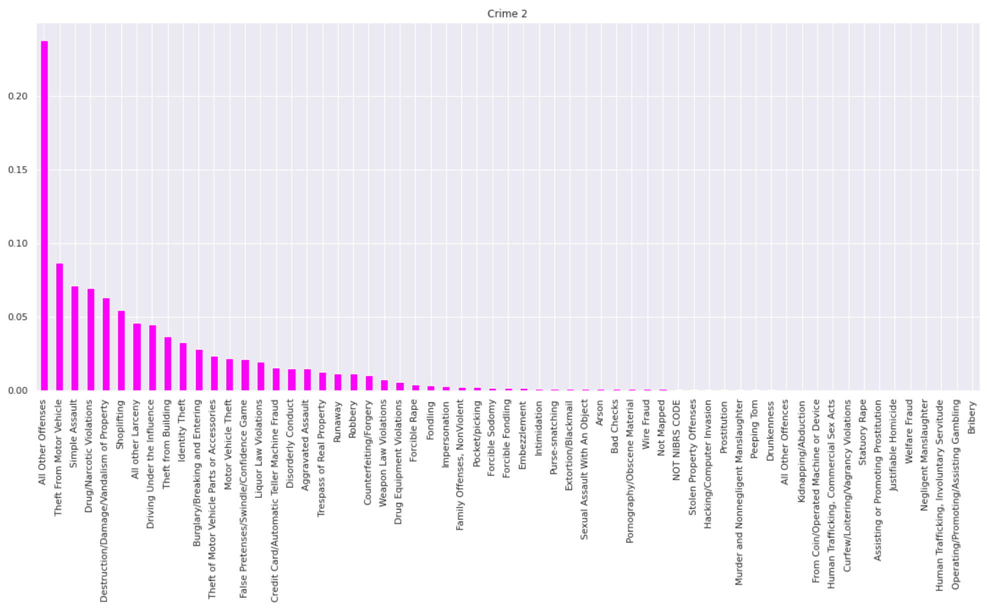

Figure 9. Crime Name 2 distribution

The value_counts for this category also show the following:

```
Crime category-2 in %:
All Other Offenses                          0.238018
Theft From Motor Vehicle                    0.087034
Simple Assault                              0.071013
Drug/Narcotic Violations                    0.069595
Destruction/Damage/Vandalism of Property    0.063211
                                              ...
Welfare Fraud                               0.000039
Negligent Manslaughter                      0.000008
Human Trafficking, Involuntary Servitude    0.000008
Operating/Promoting/Assisting Gambling      0.000004
Bribery                                     0.000004
Name: Crime Name2, Length: 61, dtype: float64
```

The most committed crime in Crime2 list is 'All Other Offenses' around 23% of the total data. These are crimes that are not categorized in a particular category. The least committed crimes in this list are 'Bribery' and 'Operating/Promoting/Assisting Gambling' around 0.00004%

Crime Name3: Crime name3 is plotted using a bar chart, figure 10, as the number of crimes in this category is too big to show in other plots.

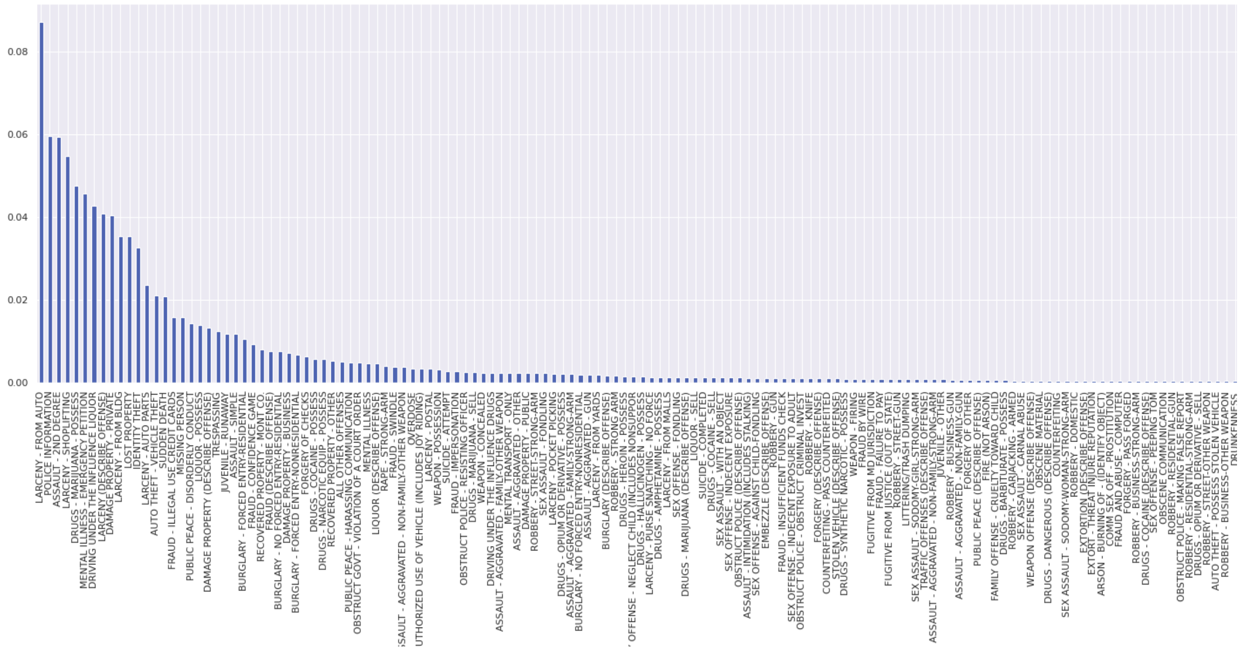

Figure 10. Crime Name 3 distribution

The value_counts for this category also show the following:

```
Crime category-3 in %:
LARCENY - FROM AUTO                               0.087124
POLICE INFORMATION                                0.059463
ASSAULT - 2ND DEGREE                              0.059401
LARCENY - SHOPLIFTING                             0.054632
DRUGS - MARIJUANA - POSSESS                       0.047500
                                                    ...
FALSE IMPRISONMENT - MINOR - PARENTAL              0.000004
OBSTRUCT GOVT - MISCONDUCT - JUDIC OFFICER        0.000004
GAMBLING - DICE GAME                              0.000004
DRUGS - OPIUM OR DERIVATIVE - SMUGGLE             0.000004
BRIBERY (DESCRIBE OFFENSE)                        0.000004
Name: Crime Name3, Length: 330, dtype: float64
```

This shows the most committed crime in this list is 'Larceny from Auto' around 8.7% of total crimes. The 'BRIBERY (DESCRIBE OFFENSE)', 'DRUGS - OPIUM OR DERIVATIVE - SMUGGLE', 'GAMBLING - DICE GAME', 'OBSTRUCT GOVT - MISCONDUCT - JUDIC OFFICER' and 'FALSE IMPRISONMENT - MINOR - PARENTAL' all of them around 0.000004%

For research Q3: For each type of crime, which places are considered the most dangerous/safest?
The grouped count bar chart allows us to show the relationship between two categorical features in one figure, figure 11. Places consist of a high number of categories in this dataset, but by creating separate data frames based on the count values of crimes in each place and using the “head ()”, and “tail()” methods, we can find the most dangerous and safest places, in which different types of crimes happen, in this dataset respectively. By using the “hue = 'Place'” on these count bar charts, we can show the values of crimes, where colours distinguish places parameter.

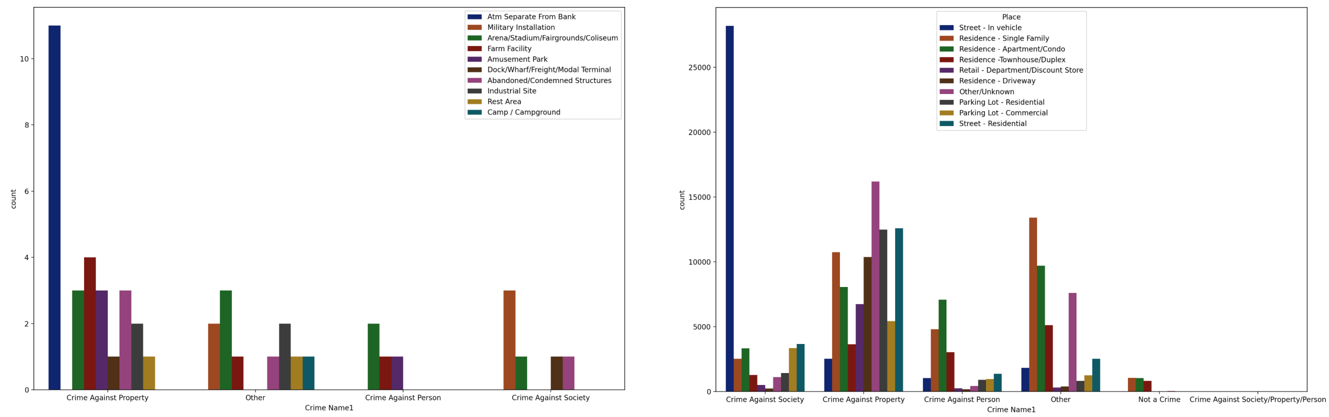

Figure 11. The most dangerous/safest places based on the type of crime

Using this plot, we can conclude that most “Crimes Against Society” occur on the street, in a person's vehicle, thus making it very important for individuals to adhere to safety protocols when they are in their vehicles. Likewise, for “Crime Against Property” besides the unknown places, streets and parking lots are considered vulnerable places. Moreover, most of the “Crimes against Person” happens in their apartments. And regarding plots for the safest places, we can see that all the places in this list are primarily private or solitary places such as farms, rest, or camp areas. Most of these places can consider safe in terms of crime against society, as well as people. But regarding the crimes against property, even in the safest places, we see considerable danger in ATMs, and the safest place is driveways.

For research Q4: Which police district deals with the highest and lowest number of crimes every year?
Getting to know the police district that deals with the highest and lowest number of crimes every year, gives an idea of the most and least busy police districts and also give insights into the districts that are the safest/most dangerous. The dataset has details of the police district name in which all the crimes are being reported. We can use this data and plot a bar chart with stacked bars, figure 12. Unlike regular bar graphs which simply represent a comparison between different categorical data based on a comparison factor, stacked bars show the comparison of related/similar categories. Using this, we can compare the count of the number of crimes that each police district is handling every year. Looking at the graph below, the 'City of

Takoma Park deals with the least number of crimes, even zero crimes in the year 2022, and the 'Silver Spring district deals with a consistently higher number of crimes every year between 2016 - 2022.

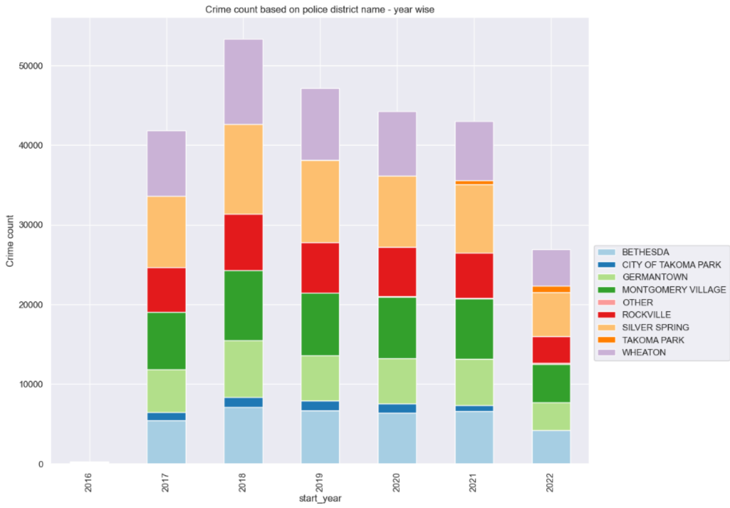

Figure 12. Crime count based on police district name - year wise

For research Q5: Which police departments are quickest to respond?

Each crime record in the dataset contains information about the start time of a crime, as well as the dispatch time of an officer. This information can be helpful to identify the average response time for the reported crimes by the different Police Districts. These details can help identify Police Districts displaying discrepancies in the time taken to act on a reported crime or vigilance. It will also create an opportunity to compare the response times of the other Police Districts and make improvements accordingly, figure 13.

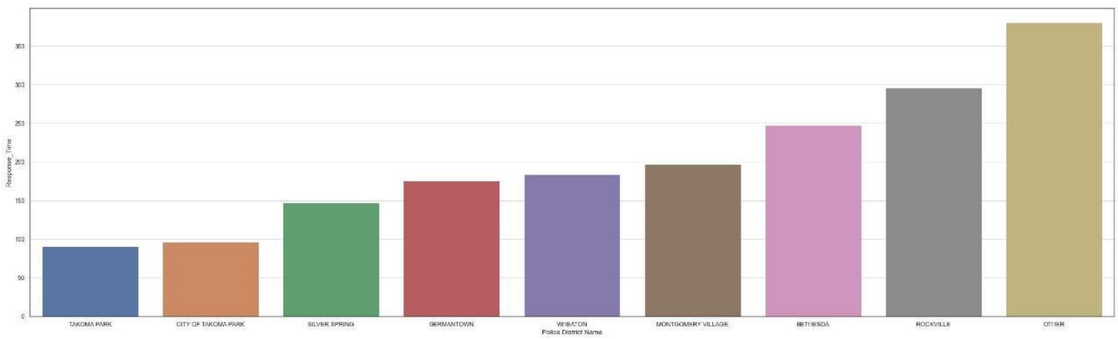

Figure 13. Response times of each police district

ROCKVILLE and BETHESDA were the slowest to respond while TAKOMA PARK and OTHER were the fastest to respond.

For research Q6: What is the frequency of crimes each year?

For this question, we are using a bar chart. Which shows us the highest frequency of crimes each year

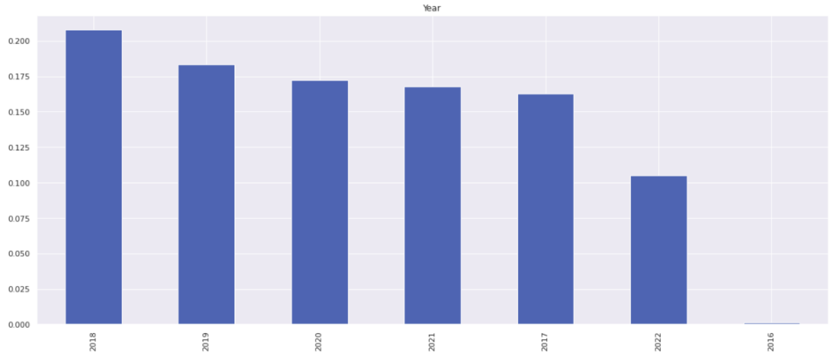

Figure 14: Frequency of crimes each year

Based on the output, the most active year in this dataset is 2018 up to 20.8% of all crimes committed in this year while the least active year is 2016 maybe because the data collection for that year only consisted of the last month.

For research Q7: Which time of the day is the most active for criminals in the county?
Understanding the most active time for criminal activity gives us an overview of what is the most active time for criminals and helps the police to be active during the peak time. To interpret this, we have split the start time column which consists of the details of what time the crime was committed. Using this column data, the time of the crime can be extracted and understand whether the time of the day is either day or night. A new column is created to identify whether the start time is day or night. This newly added column is used to plot a bar chart which shows a comparison between the count of crimes being committed during day/night, figure 14.

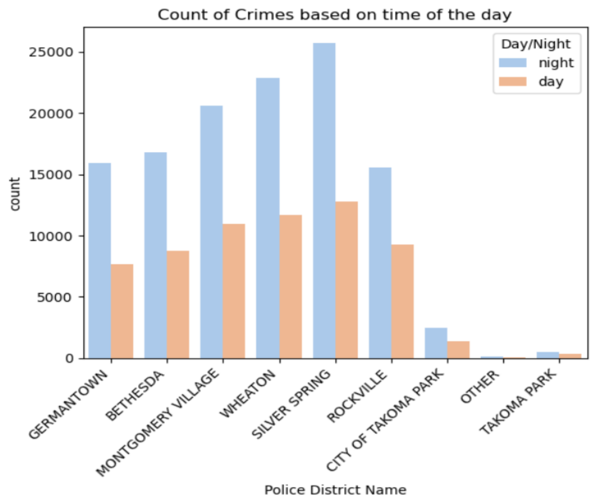

Figure 15. Count of crimes based on time of the day

For research Q8: What are the hot spot locations for criminal activities in Montgomery County?
An overview of areas in the Montgomery County that are being targeted by criminals can be a crucial visualisation for the Police and Government Officials. It can help to dispatch forces for patrolling in the hotspots, make strategies or analyse the reports further. Additionally, it provides colour gradient contours of such on geographical maps to clearly identify the locations along with crime rates where red colour identifies such hotspots, figure 15.

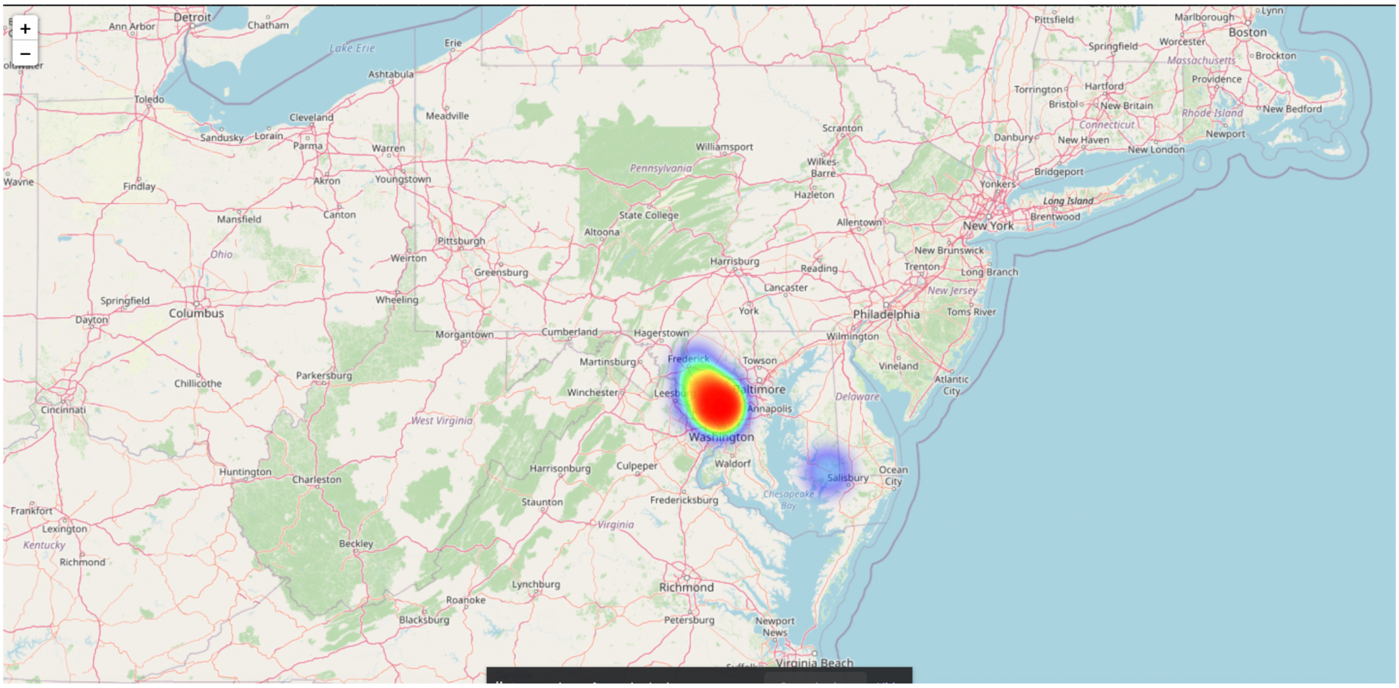

Figure 16. Hotspots of criminal activity in Montgomery County

For research Q9: What is the relationship between the weekdays and the number of crimes?
A bar chart can be used to explain the slight difference between crime rates for each weekday, figure 16. Most crimes occur on Fridays, as it is the day before the weekend when most people stay out late. We have the least number of crimes on Sundays, too, because Sunday is the first day before the week begins, so people go home earlier to prepare for the week. It is also common for shops and recreational places to close early on Sundays.

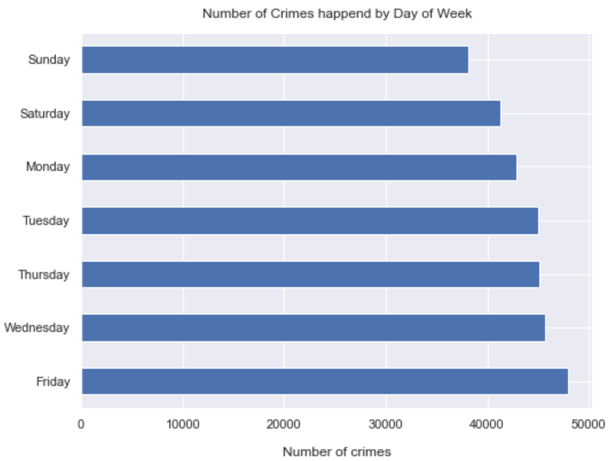

Figure 17. Count of crimes based on the day of the week

For research Q10: Which cities are considered the safest and the most dangerous ones in terms of the number of crimes?

Each crime record in the dataset is accompanied by the city where it occurred. As a result of the records gathered in the defined period, this question aims to find the most dangerous and safest cities in the dataset. We can create an array of city names and the recorded crimes for each by using the value_counts () method,

and convert it to a data frame by using to_frame(). To visualise this information, we can use a bar chart, to plot the counted values for each city, figure 17 and figure 18.
First, we use the sort_values on the created data frame based on the number of crimes, and then we use bar plot to visualise the most dangerous and safest cities.

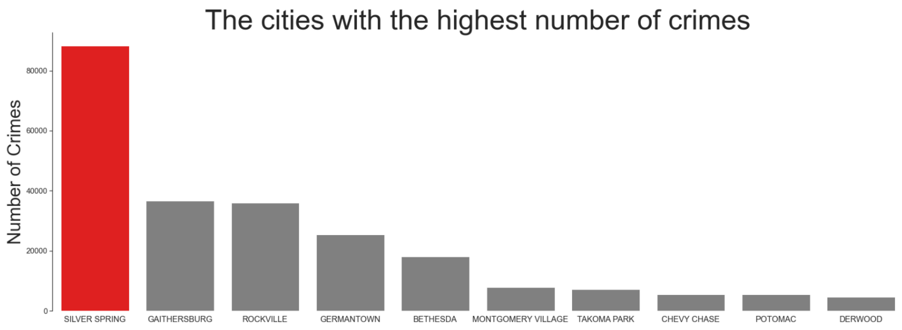

Figure 18. Cities with the highest number of crimes

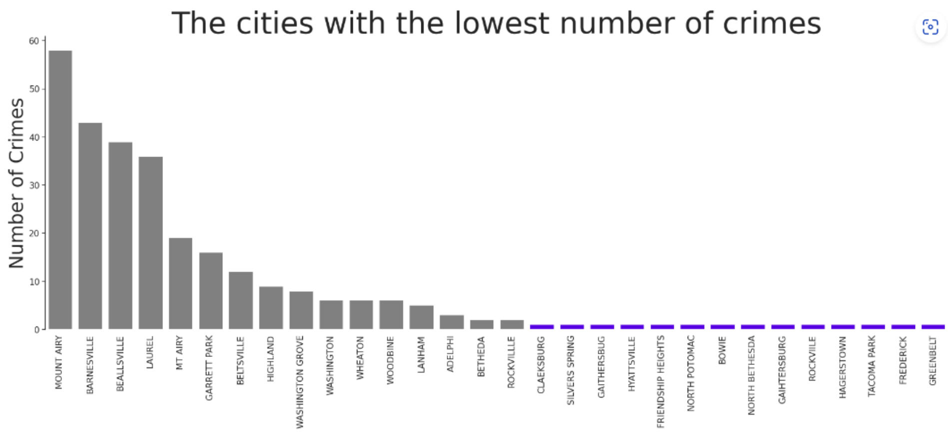

Figure 19. Cities with the lowest number of crimes

As it is presented SILVER SPRING, which is in red, has the greatest number of crimes in the dataset, and can be considered the most dangerous city in Maryland county.
Likewise, in finding the safest city, we figure out there are multiples cities with the minimum number of crimes, just one record, to find the list of those cities we can define ["City”] ==1 condition, in the created data frame with value_counts () method.
As a result, there are 14 cities that can be considered the safest cities in this area, to have a better visualisation, to compare the numbers in the safest cities we are going to plot the 30 safest cities in Maryland county. These 14 cities are shown in blue, in the following bar plot.[5]

V. Recommendation & Conclusion

A. *Recommendations*

- People should be cautious while driving, whether on the street or in parking lots, as well as at ATMs. Additionally, municipalities can think about providing more security services and tools for these places, such as installing CCTVs, to reduce property and society crimes.
- From the plot Silver Spring district deals with a higher number of crimes, when compared to other districts. The recommendation is to improve the security and vigilance in the operating areas of districts with high crime rates like Silver Spring, Wheaton, Montgomery Village, etc.
- Based on the observations, the crime rate is high at night compared to the number of crimes during the day across all the police districts. It would be better if surveillance and patrolling is increased during the latter part of the day so that there is a sense of caution among criminals.
- Regarding the weekdays, as Fridays are more dangerous maybe some rules to prevent committing a crime in target places regarding question 3, can be considered just for Fridays.
- As SILVER SPRING city has a considerable number of different crimes than other cities, new strategies are needed to control the crime rate. As a role model, municipalities can study policies in the safest cities such as WHITE OAK to decrease these numbers.

B. *Conclusion*

After performing a thorough analysis of the dataset, we were able to come up with visualizations that help us understand the hidden patterns, outliers, and trends in the data. The visualizations of the research questions give us insights into the criminal activities in the county. The visualizations show the safest/the most dangerous cities, and police districts, what time of the day is most active, the number of victims being affected, etc. When the dataset is large and contains a lot of numerical data, plotting graphs give us a pictorial representation and this works to get an idea of various questions or patterns from the data. After performing Initial Data Analysis and cleaning up null/missing values, a few suggestions can be given to make sure that information is maintained properly without leaving empty fields and appropriate data is being entered. Based on the visualizations used to answer these research questions, recommendations on when/where police can look out for criminals who are the most active within Montgomery County are given. This will help the police force to look out for patterns on when they must be alert and what appropriate actions can be taken to reduce criminal activity.

VI. References

[1] "National Incident-Based Reporting System (NIBRS)," Bureau of Justice Statistics. [Online]. Available: https://bjs.ojp.gov/data-collection/national-incident-based-reporting-system-nibrs#publications-0. [Accessed: 10-Nov-2022].

[2] Conlen, B.M. (no date) Kernel Density Estimation, Redirect to... mathisonian.com. Available at: https://mathisonian.github.io/kde/ (Accessed: December 7, 2022).

[3] Seaborn.histplot# (no date) seaborn.histplot - seaborn 0.12.1 documentation. Available at: https://seaborn.pydata.org/generated/seaborn.histplot.html (Accessed: December 7, 2022).

[4] Appendix D - USPS state abbreviations and FIPS codes (2005) U.S. Bureau of Labor Statistics. U.S. Bureau of Labor Statistics. Available at: https://www.bls.gov/respondents/mwr/electronic-data-interchange/appendix-d-usps-state-abbreviations-and-fips-codes.htm (Accessed: December 7, 2022).

[5] Kosourova, E. (2022) How to plot a bar graph in Matplotlib (W/ 9 examples), Dataquest. Available at: https://www.dataquest.io/blog/how-to-plot-a-bar-graph-matplotlib/ (Accessed: December 7, 2022).