Protecting neighborhood through data science

Bayarbat Bayarsaikhan

10/19/2022

Outline

1. **Introduction**
    - Business Problem
    - Target audience
2. **Data section**
    - Data collection
    - Data wrangling
3. **Methodology**
4. **Results**
5. **Conclusion**

1: Introduction

According to study done 2002 by Sherman and Eck, visible law enforcement officer can reduce the crime significantly in "hot zones" where crime is concentrated. Since law enforcement officers are limited resource it is beneficial to find out the best places to deploy the officers.

1.1: Business problem

The city chosen to do the study is Chicago a city in U.S state Illinois. Chicago has a population of 2.7 million and crime rate of 3926 per 100,000 people in 2020. Meaning approximately 100000 crimes were committed during just 2020. In light of this insight, I would like to possible help government/city with understanding town better and improve the neighborhoods

1.2: Target audience

- Government and police department who wants to improve the city
- Business owners who wants open a new venue in safer areas of the city

2: Data section

2.1 Data collection

First of all we need information on Chicago city neighborhood districts, latitude, and crime rate. We collected the data from Chicago citys database: https://data.cityofchicago.org/Public-Safety/Crimes-2001-to-Present/ . We can check that dataframe contains what we need.

```
#Initial data count
df_filtered.count()
```

```
ID                      207905
Case Number             207905
Date                    207905
Block                   207905
IUCR                    207905
Primary Type            207905
Description             207905
Location Description    207063
Arrest                  207905
Domestic                207905
Beat                    207905
District                207905
Ward                    207894
Community Area          207905
FBI Code                207905
X Coordinate            202902
Y Coordinate            202902
Year                    207905
Updated On              207905
Latitude                202902
Longitude               202902
Location                202902
dtype: int64
```

2.2: Data wrangling

First we clean up the rows without any input. As you can see we almost dropped 5000 rows and now the columns have consistent amount of entry

In [5]:
```
#Remove any row that contains NaN
df_filtered = df_filtered.dropna(axis=0)
#Data count after clean up
df_filtered.count()
```

Out[5]:
```
ID                    202264
Case Number           202264
Date                  202264
Block                 202264
IUCR                  202264
Primary Type          202264
Description           202264
Location Description  202264
Arrest                202264
Domestic              202264
Beat                  202264
District              202264
Ward                  202264
Community Area        202264
FBI Code              202264
X Coordinate          202264
Y Coordinate          202264
Year                  202264
Updated On            202264
Latitude              202264
Longitude             202264
Location              202264
dtype: int64
```

2.2 Data Wrangling

- We can also clean the date column and remove the exact hour and seconds so it easier to group the data by date for visualization. Now the date input is just year, month, and date.

```
#Clean up the datetime so it is easier to group
from datetime import datetime
for index, row in df_filtered.iterrows():
    date = row["Date"]
    dateo = datetime.strptime(date, '%m/%d/%Y %H:%M:%S %p')
    dateo = dateo.date()
    df_filtered.loc[index, 'Date'] = dateo
df_filtered.head()
```

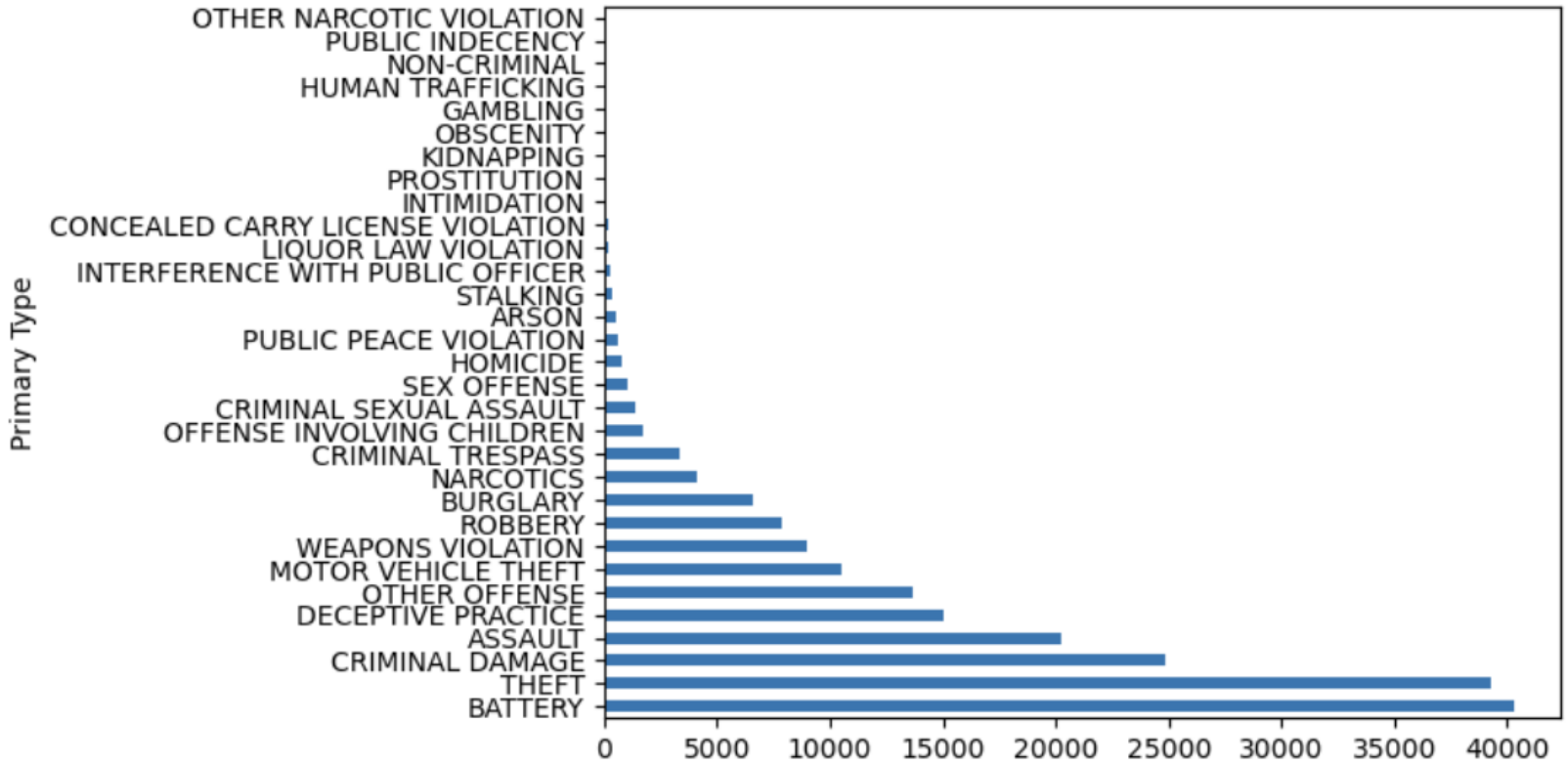
	ID	Case Number	Date	Block	IUCR	Primary Type	Description
265	12571973	JE482457	2021-12-19	042XX S MOZART ST	0460	BATTERY	SIMPLE
63535	12602803	JF125633	2021-10-21	083XX S STONY ISLAND AVE	500E	OTHER OFFENSE	EAVESDROPPING
69369	12540388	JE444591	2021-11-14	086XX S COTTAGE GROVE AVE	0850	THEFT	ATTEMPT THEFT
69760	12541139	JE445494	2021-11-14	034XX W 38TH ST	0486	BATTERY	DOMESTIC BATTERY SIMPLE
78119	12540496	JE444717	2021-11-14	070XX S INDIANA AVE	0820	THEFT	$500 AND UNDER

3: Methodology

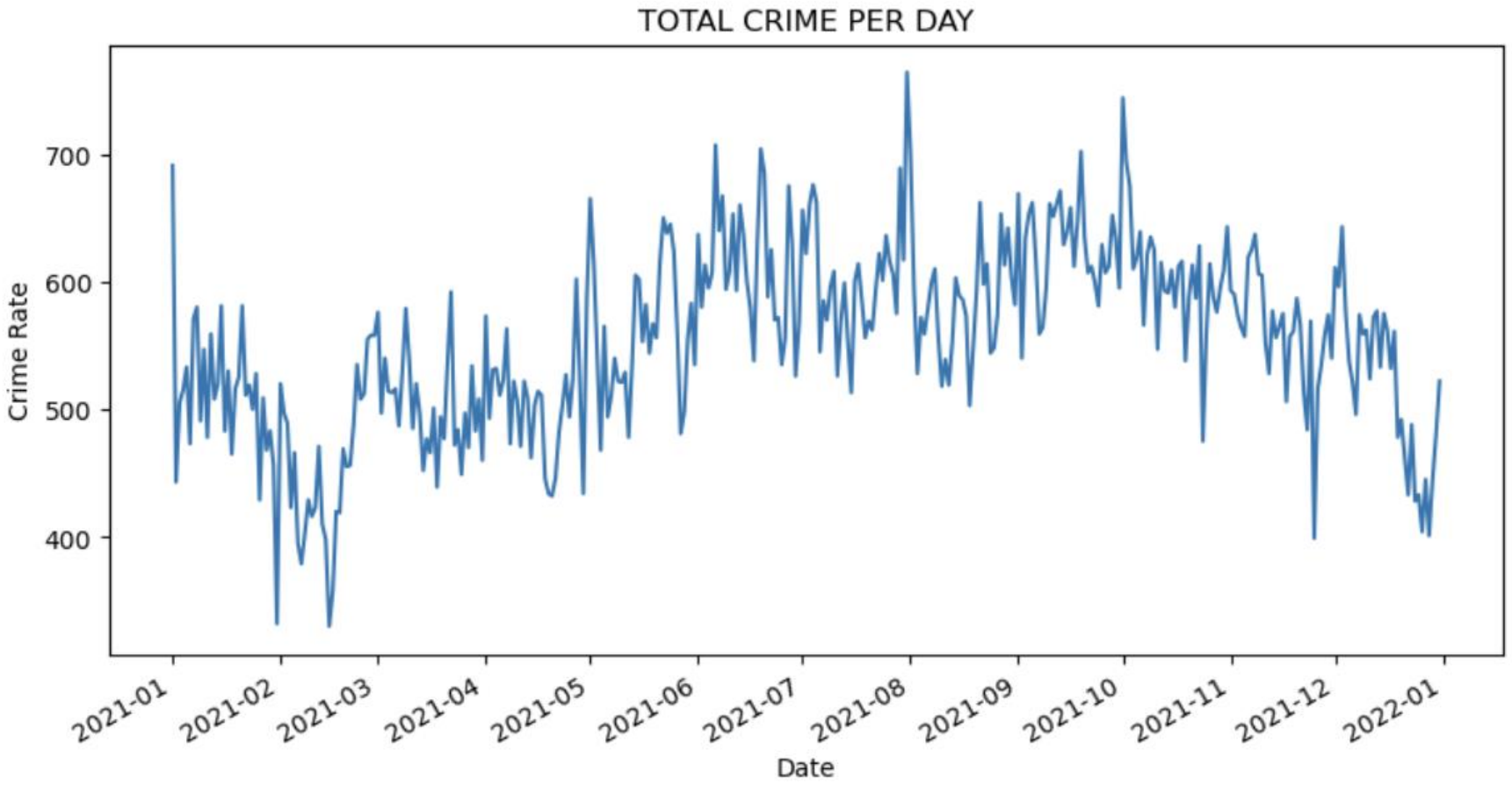
3.1: Graph and chart using MatPlotLib

Most common type of crime

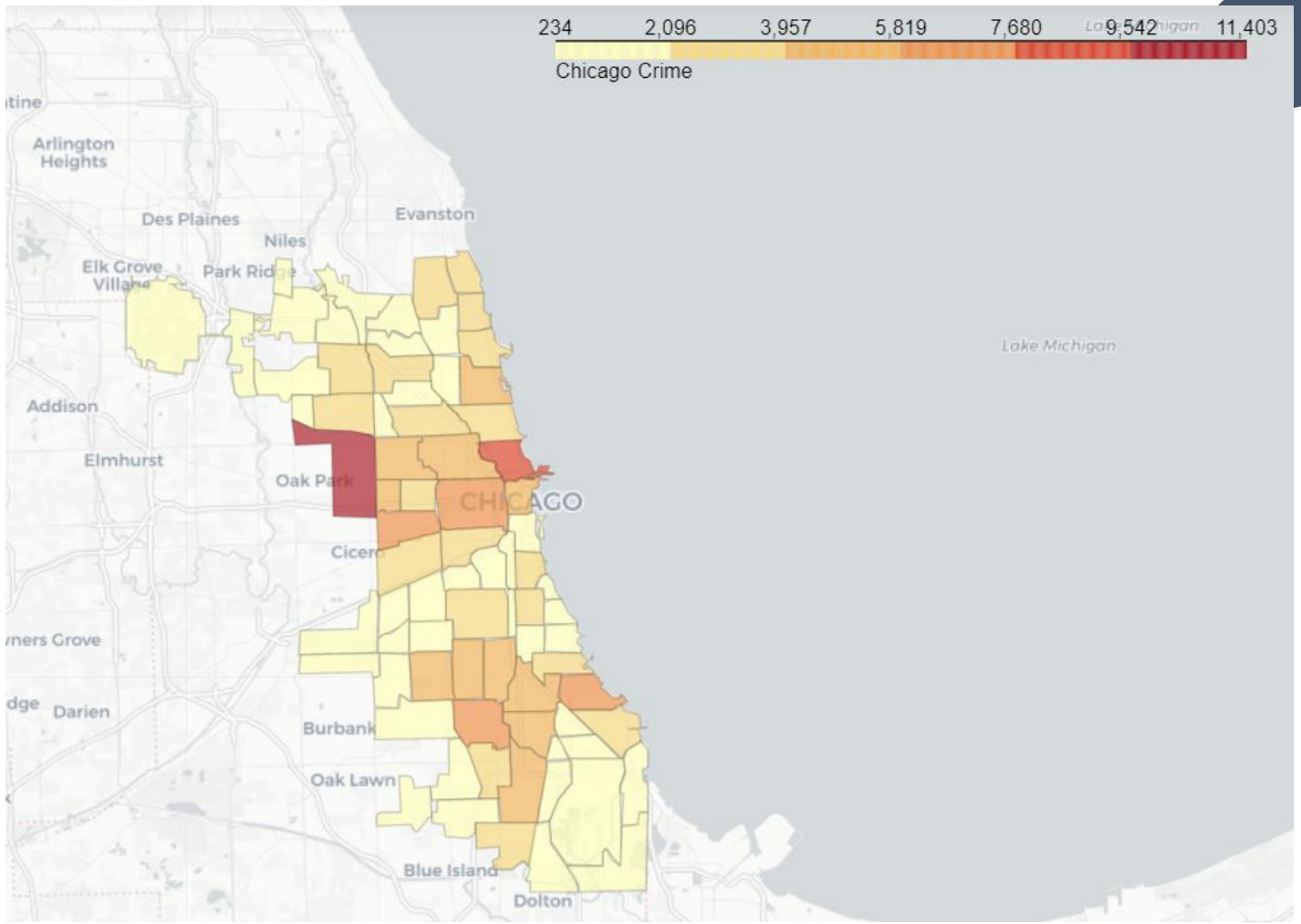
```
#Bar chart of type of data and crime rate during 2021
ax = df_type.plot.barh()
```



3.1: Graph and chart using MatPlotLib



3.2: Heat map using Folium and Geojson

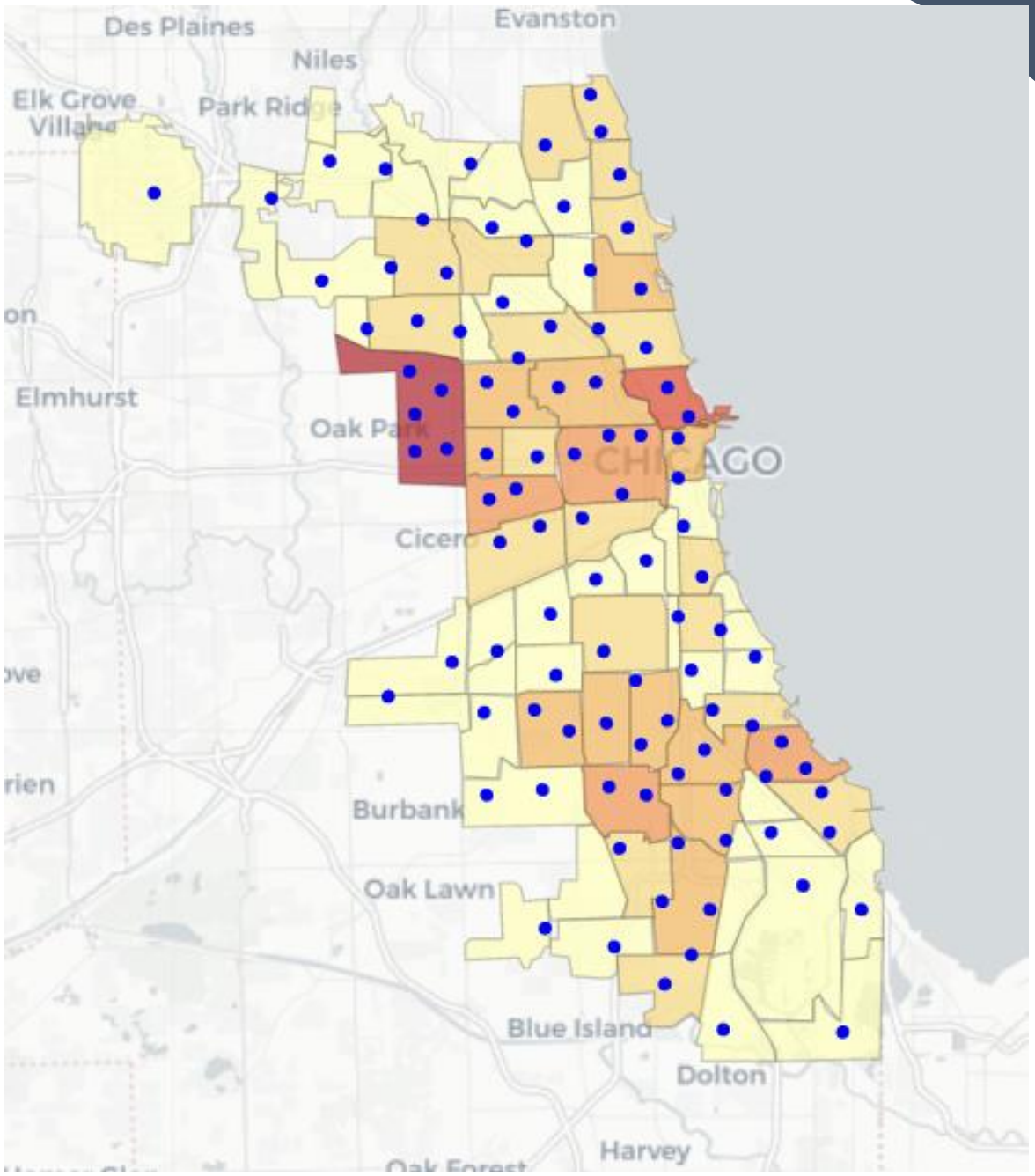


3.3: Clustering using KMean

- We can find the best position to deploy law enforcement officer depending on how many officer is available using KMean clustering and assign each cluster to police officer. Since the crime data has been clustered together through location we can use this model in the future to find the nearest and assign law officer to new crime

4: Results

As we can see from heat map and clustering areas: areas in middle of the city that crime occurs often has most clustering. This module can be used not only for optimal place for police officers, it can also be used to determine which officer is closest to new crimes



5: Conclusion

Chicago is an international city with many crimes happening everyday and I think we have gone through the process of identifying the business problem, specifying the data required, clean the datasets, performing a machine learning algorithm using k-means clustering and providing some useful tips to our stakeholder. As for future development we can create predictive module that can not only guess where he future crime will happen but also when.

Thank you

Bayarbat Bayarsaikhan

10/31/2022