Data Analysis on Chicago Crime Dataset

1.Introduction

Working with data is all about gaining knowledge. Whether that knowledge is consumed by a person or acted on by a data product(ML system), our goal as professionals working with data is to use observations to learn about how the world works. We want to turn information into insights, and asking the right questions ensures that we're creating insights about the right things. This is what we hope to achieve in this document.

This was an opportunity to dig into one of the most famous open dataset known as Chicago Crime dataset. There's been a lot of resources available on the web describing and analyzing this data, by which quite a few of them provide insightful analyses. In this short document, I do my best to detail out my own approach, won't shy away from challenges I faced and tools I used. Before going any further, I would like to thank my colleague Romina Shiri Bsc. for sharing with me her own work on the subject, and Fatemeh Haji Ahmadi PhD, for her advises and patience to make this happen.

City of Chicago is the most populous city in the U.S. state of Illinois, and the third-most populous city in the United States, following New York City and Los Angeles. With a population of slightly less than 9 millions as of 2022, it is also the most populous city in the Midwestern United States and the fifth most populous city in North America. Located on the shores of freshwater Lake Michigan, Chicago was incorporated as a city in 1837 near a portage between the Great Lakes and the Mississippi River watershed and grew rapidly in the mid-19th century. Chicago is an international hub for finance, culture, commerce, industry, education, technology, telecommunications, and transportation. Chicago's 58 million tourist visitors in 2018 set a new record, and Chicago has been voted the best large city in the U.S. for four years in a row by Condé Nast Traveler. The city was ranked first in the 2018 Time Out City Life Index, a global urban quality of life survey of 15,000 people in 32 cities, and was rated second-most beautiful city in the world (after Prague) in 2021. But also where there's this amount of visitors and tourists in a city, it might be a bigger chance for criminals (e.g. burglars) to show up more often, and Chicago is no exception. Checking out its crime statistics, Chicago had a murder rate of 18.5 per 100,000 residents in 2012, ranking 16th among US cities with 100,000 people or more . Violent crime rates vary significantly by area of the city, with more economically developed areas having low rates, but other sections have much higher rates of crime. As of 2021, Chicago has become the American city with the highest number of carjackings. Chicago began experiencing a massive surge in carjackings after 2019, and at least 1,415 such crimes took place in the city in 2020. According to the Chicago Police Department, carjackers are using face masks that are widely worn due to the ongoing COVID-19 pandemic to effectively blend in with the public and conceal their identity. Checking the validity of these claims using the dataset

can itself be a useful practice. Nevertheless, diving into the data ourselves, I separated the main components of this humble analysis into the following parts:

- Business Question
- Data Collection
- Data Exploration
- Data Preparation
- Data Modeling
- Model Evaluation and Interpretation
- Conclusion and Communicating Results

Now without further ado, we get right into each one of them.

2. Data Analytics

2.1. Business Question

Without knowing exactly where we're going our attempt might result in vein. So we must first understand what we're dealing with and what our final result might end up. We're interested in designing an intelligent predictive modeling, making predictions on whether a piece of record will be violent or not? This can be quite useful in a couple of aspects. It will first, informs us whether we prepare ourselves for making allocation for other Emergency services, namely, healthcare facilities to provide sufficient resources. Additionally our police officers assigned to the task at hand get a useful idea on what to expect at the incident.

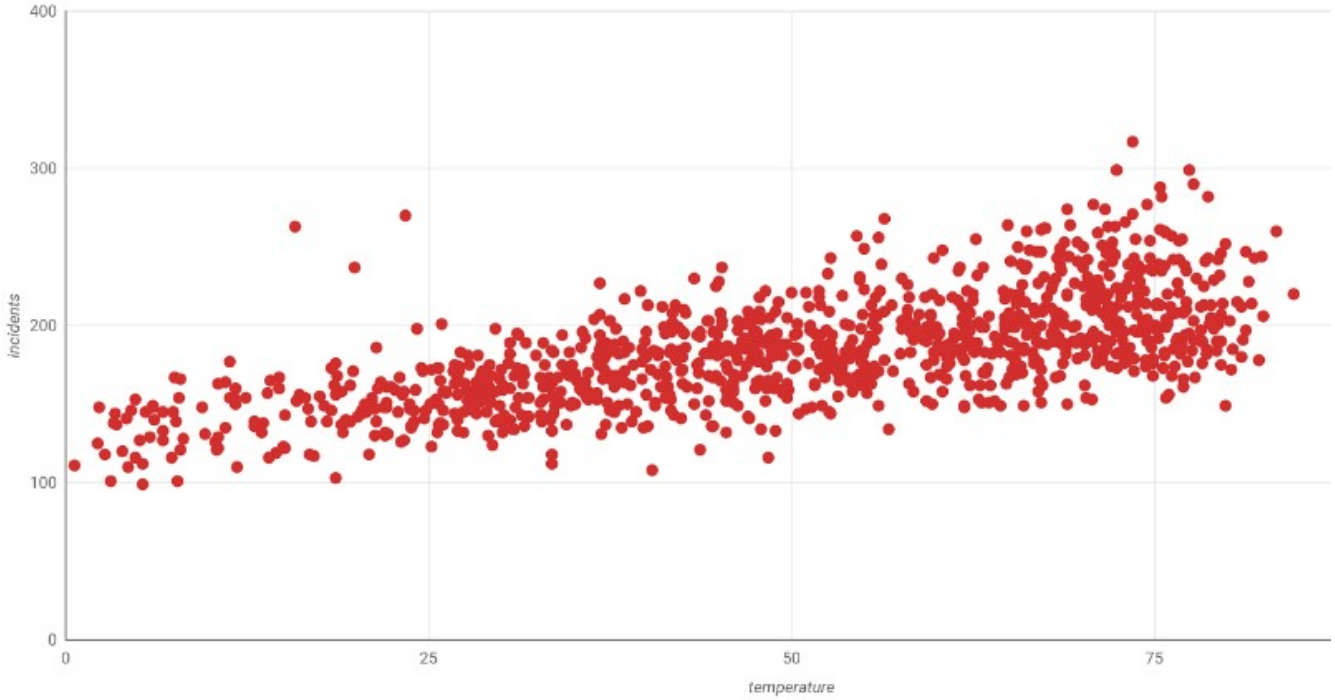
2.2. Data Collection

The next stop is to provide relevant data for the task at hand. The dataset chosen for this project consists of incidents of crime reported in the city of Chicago from 2001 to present. Data is extracted from the Chicago Police Department's CLEAR (Citizen Law Enforcement Analysis and Reporting) system. It is one of the richest data sources in the area of crime. The dataset includes enough information about Date, Type, Description, location etc about the crime for our analysis. Most people get this dataset from Kaggle. Unfortunately using Kaggle website your only option is to query through it using BigQuery and get your data. This proved to be effective when you want to learn about BigQuery or want to make a sample out of data to make some quick descriptive analytics. However to get the whole data I visited the portal of city of chicago, which provided the latest dataset, it also turned out to be particularly useful if you want other data sources with regard to Chicago, which we'll discuss more in detail later. Additionally, just like BigQuery, you can use yet another wonderful package called Socrata which is THE hub for using huge collection of datasets around the world. It

provides API endpoints for developer working with historically recognized and scientifically valued datasets. However the free tier restricts the amount at which you can request from. I used it for updating my downloaded dataset which I uploaded into my Google account so I can always have the latest version of dataset.

Challenge #1 Building a program to update the dataset using Socrata API

Other data sources would definitely provide additional insights, like economical and educational data of the city of Chicago. For example consider this beautifully done visualization that investigated the relationship between weather temperature and crime incidents.



So just like above, a good collection of other external data sources might provide useful insights, and ultimately might as well improve our predictive analytics.

Challenge #2 Providing other external data sources like temperature

The dataset contains over 7 million records of the crime. Data of this size needs fast and efficient data processing. Though I used vanilla pandas to the analysis, I believe Spark framework would be a much appropriate choice, as its in-memory processing capability makes it easy to deal with data of this volume.

Challenge #3 Handling the size of the dataset ~1.6GB

Dataset by itself was huge, so without getting sophisticated tools like Spark DataFrame Vaex, or Dask loading the dataset directly into the RAM would indeed crash a 12GB

RAM machine. However tweaking with a few tricks on chunks of data instead of the whole proved to be effective, where I could successfully reduced the size of the dataset to ~600MB, this way I could load the dataset into my own machine, handling manipulation and exploration almost smoothly.

Columns which I ended up (and I must confess that I carefully got rid of a few of them)

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 7490070 entries, 0 to 7490069
Data columns (total 19 columns):
 #   Column               Dtype
---  ------               -----
 0   Date                 datetime64[ns]
 1   Block                object
 2   IUCR                 object
 3   Primary Type         object
 4   Description          object
 5   Location Description  object
 6   Arrest               bool
 7   Domestic             bool
 8   Beat                 int16
 9   District             int8
 10  Ward                 int8
 11  Community Area       int8
 12  FBI Code             object
 13  Year                 int16
 14  Latitude             float64
 15  Longitude            float64
 16  Month                int8
 17  Day                  int8
 18  Hour                 int8
dtypes: bool(2), datetime64[ns](1), float64(2), int16(2), int8(6), object(6)
memory usage: 600.0+ MB
```

- **Date** is a Timestamp of the incident, columns **Month**, **Day**, **Hour** are taken directly from this column. We can also add extra time-based columns, like Week, Day of the Week, etc.
- **Block** is a partially redacted address where the incident occurred, placing it on the same block as the actual address.
- **IUCR** stands for Illinois Unifrom Crime Reporting. This code is directly linked to the Primary Type and Description. See this link for more information.
- **Primary Type** is the primary description of the IUCR code mentioned above.
- **Description** is the secondary description of the IUCR code, a subcategory of the primary description.
- **Location Description** is description of the location where the incident occurred.
- **Arrest** shows whether an arrest was made.
- **Domestic** shows whether the incident was domestic-related as defined by the Illinois Domestic Violence Act.
- **Beat** indicates the beat where the incident occurred. A beat is the smallest police geographic area – each beat has a dedicated geographic territory(See this link for more info).
- **District** indicates the police district where the incident occurred.
- **Ward** refers to a number labeling City Council district where the incident occurred.
- **Community Areas** indicates the community area where the incident occurred. Chicago has 77 community areas.

- **FBI Code,** the crime classification as outlined in the FBI's National Incident-Based Reporting System (NIBRS).
- **X-Coordinate, Y-Coordinate, Latitude, Longitude, Location(just (Lat, Lon)), Year, Updated On**

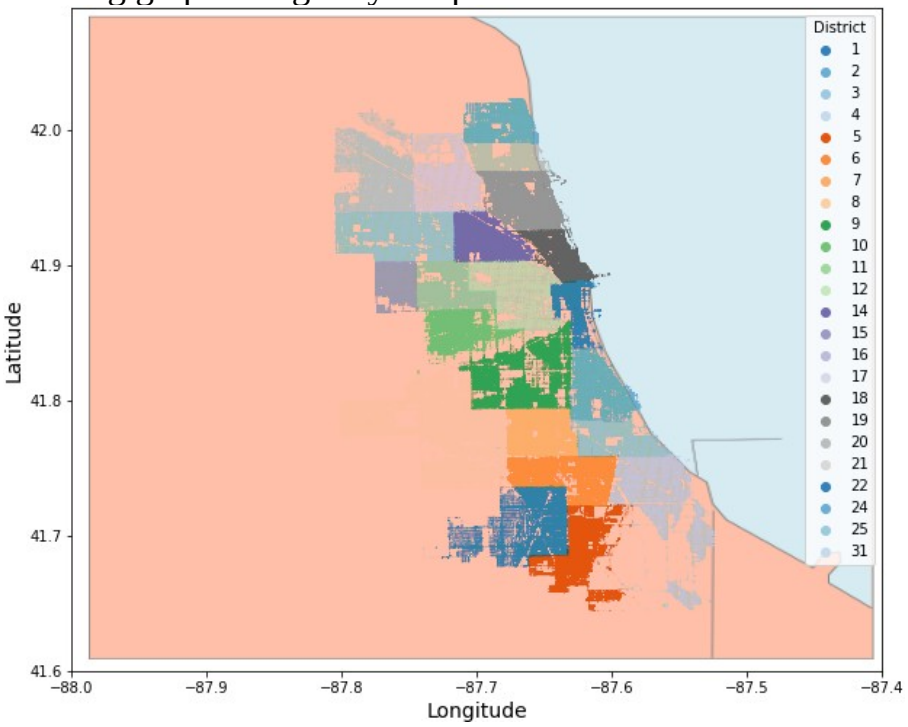
2.3. Data Exploration

Data Exploration commonly known as Explanatory Data Analysis or EDA is considered to be one of the core activities involved in any data science project of any level. With no exception for this analysis, we provide a few useful "descriptive" info on the data at hand, with the hope of detecting underlying patterns or relationship between predictors and their ultimate relationship with the target(**Domestic**) itself.

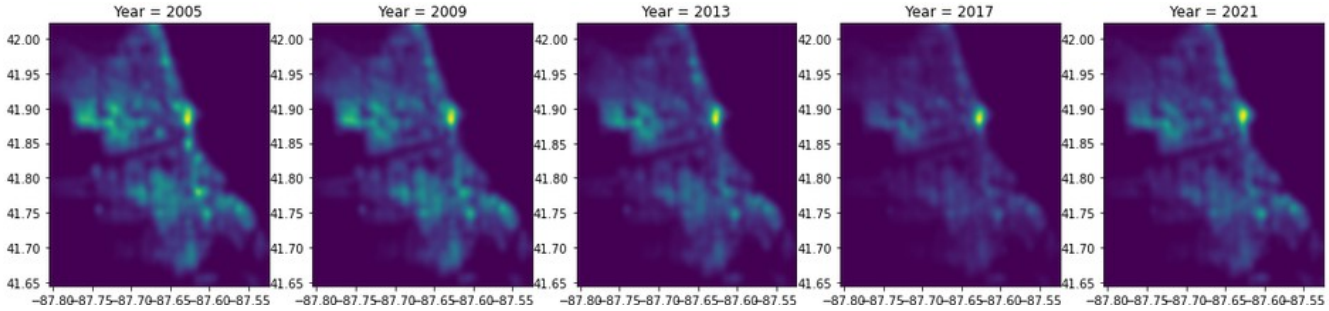
Again, considering the size of the dataset, it was almost impossible to work with all these points on a go-to geographic python libraries like Folium, Mapbox, Basemap, etc.

Challenge #4 Is it possible to show all the geographical points on the map?

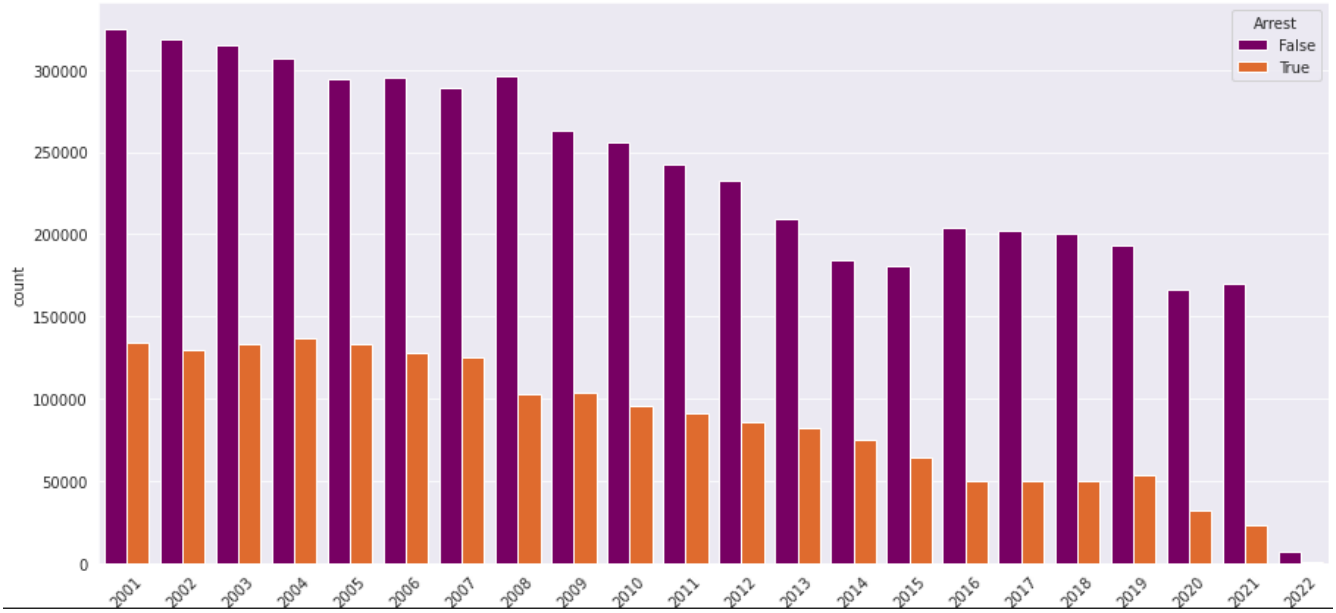
It turns out if you manage to somehow create a snapshot of a map, putting that as a background and create a basic scatter plot using matplotlib library (this requires carefully measuring the edges' positions of the background image), you might get something after all. Creating the background image as the place of Chicago on the map using Basemap, combining that with knowing data points' police district, we were able to create the following graph using only matplotlib.

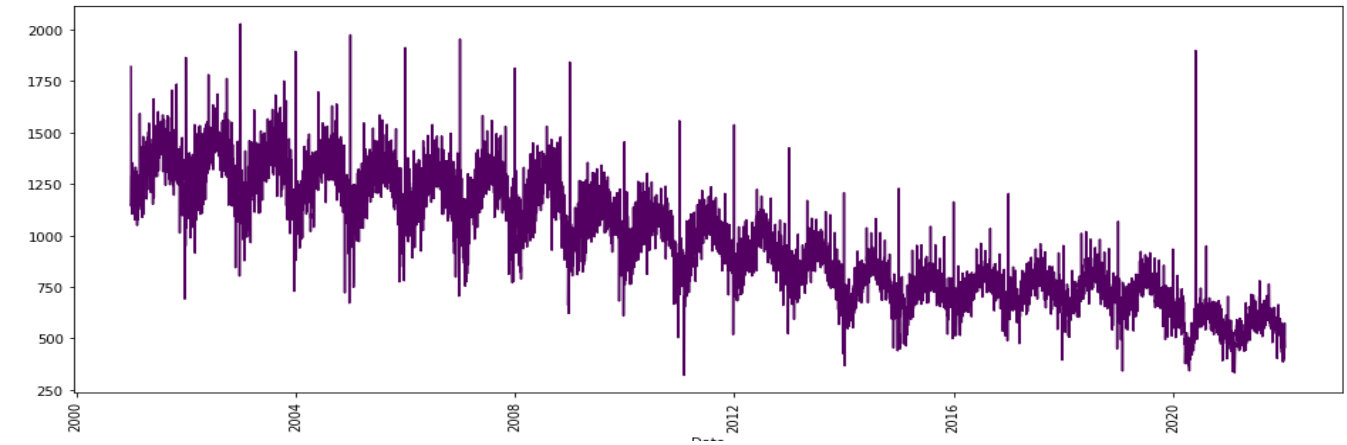


Checking the concentration of crimes on the map we see a recurring pattern in the city:



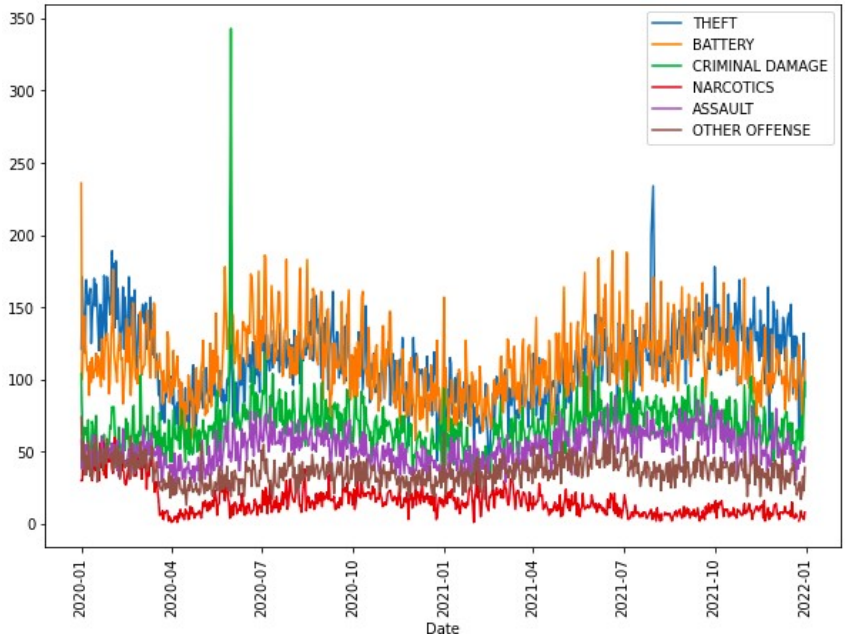
We can spot a very bright area at the west coast side of the city, actually that's a known crime-prune known as Chicago Loop. We can investigate more into checking what kind of crimes happened mostly in that area, or checking whether these crimes are arrest-related or not and so forth. Also observe that the overall bright areas of crimes are decreasing over the year. The following plot confirms this claim.

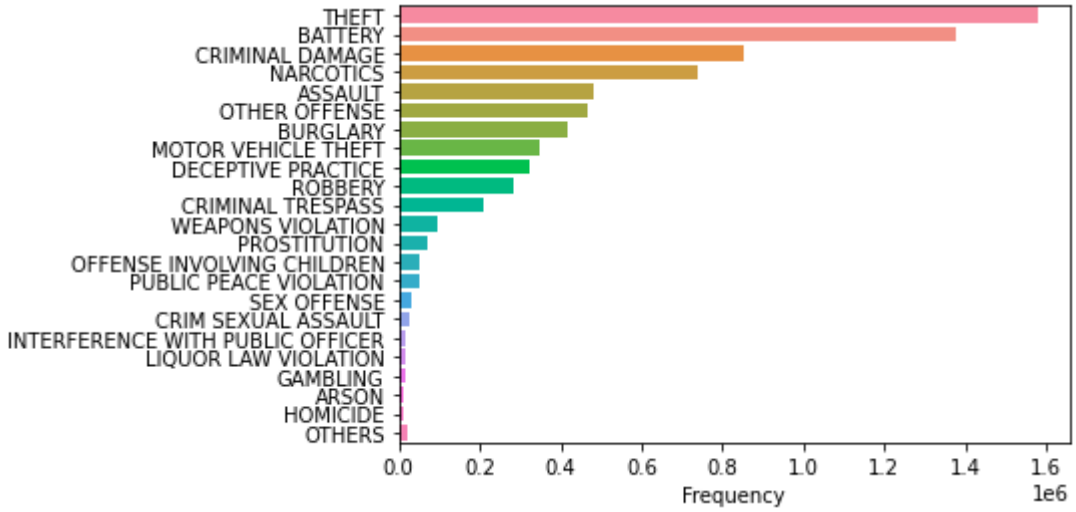




Although the decreasing behavior is not monotonic, but the overall decline is plain obvious. We can also detect seasonality, trend, and an unusually spike at the right end when checking the frequency of data over time.

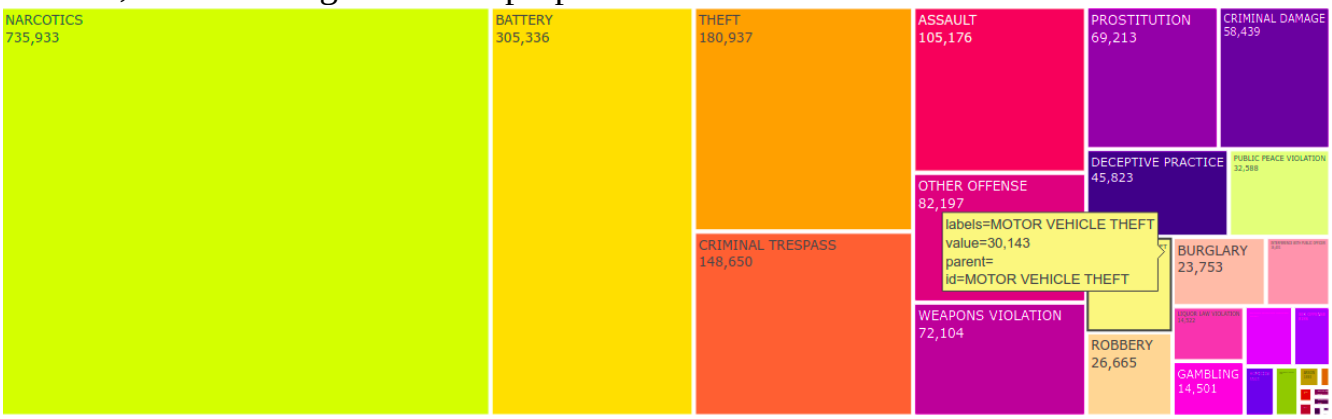
It turns out this spike is somehow related to the series of protests and civil unrest against police brutality and racism caused by the murder of George Floyd in May 26th 2020. The following graph also confirms it:
Checking the frequencies of crimes' types, actually whether an arrest was made or not plays a crucial role into the frequencies as the below graphs show:



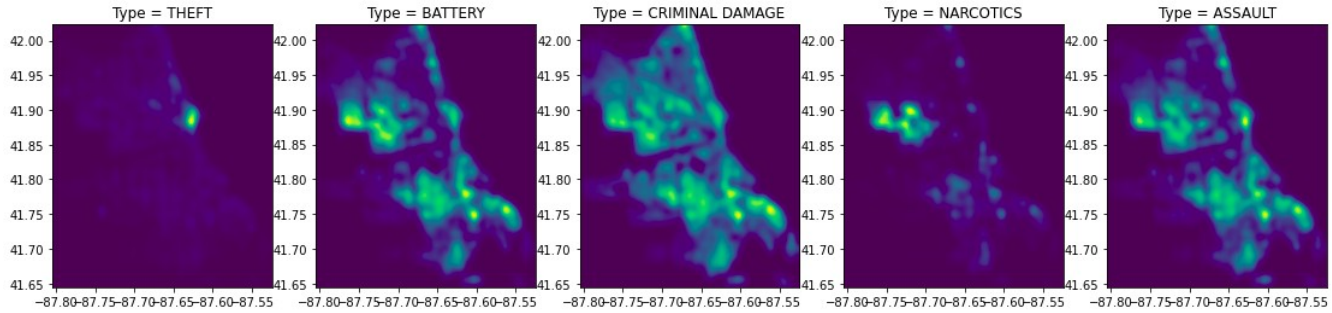


This is a naive count plot, and depending on the context some categories might point to the same underlying class(for example SEX OFFENSE and CRIM SEXUAL ASSAULT might point to the same label or class), also there's seen some typical human errors which haven't been handled carefully(for example both CRIM SEXUAL ASSAULT and CRIMINAL SEXUAL ASSAULT have been seen on the Primary Type column, while in our demonstration, we made it into OTHERS which might be better to be considered in a more serious settings).

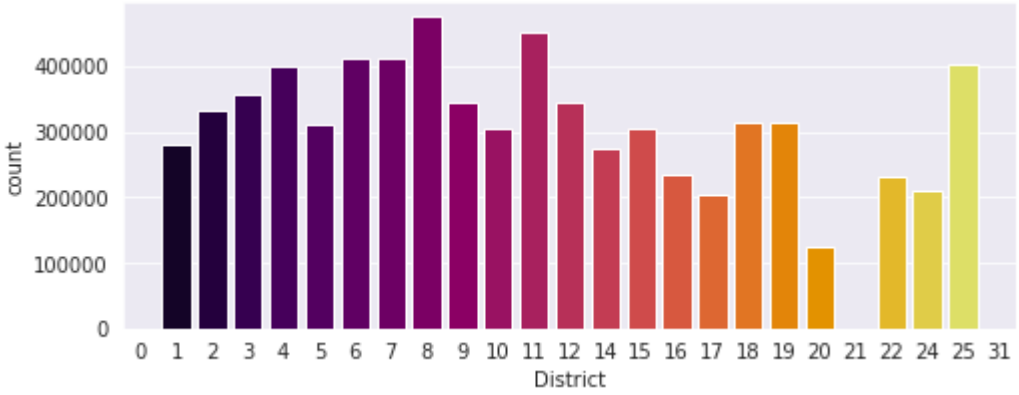
Using treemaps also turn out to be very visually informing taking the proportionals into account, the following shows the proportion of crimes which result in arrests:



We can also check the concentration of each particular crime on the map, for example for the five most occurring crimes we see, for example, that crimes related to THEFT, the famous Loop area is very bright, however for NARCOTICS-related crimes west side area is more involved. These kinds of concentrations over the map might as well come in handy when we're dealing with modeling, in particular predictive modeling.



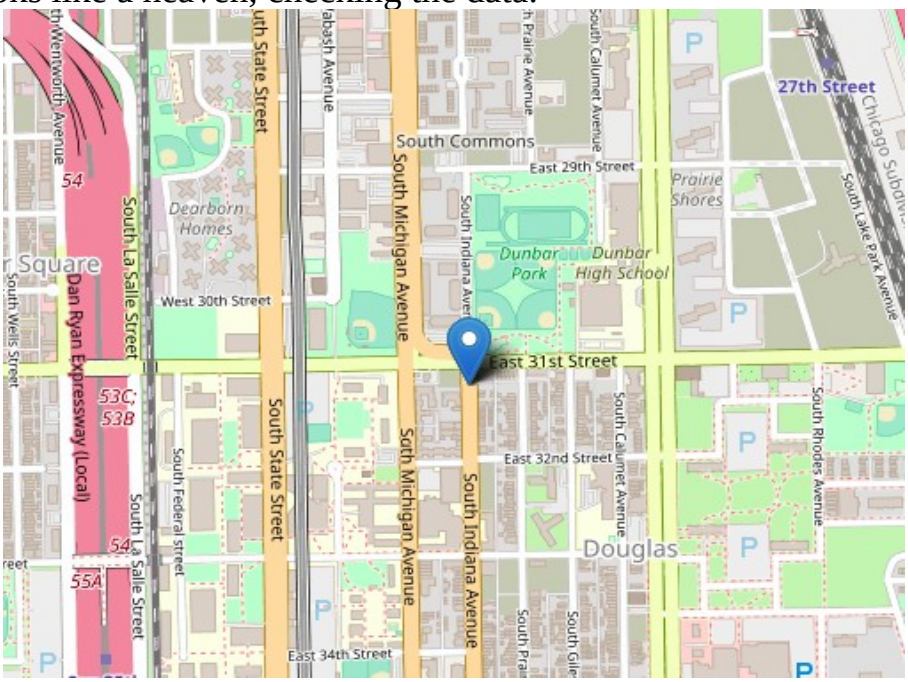
As you can see we can go further and investigate into other variables as well. In face, let's check Police Districts frequency bar chart:



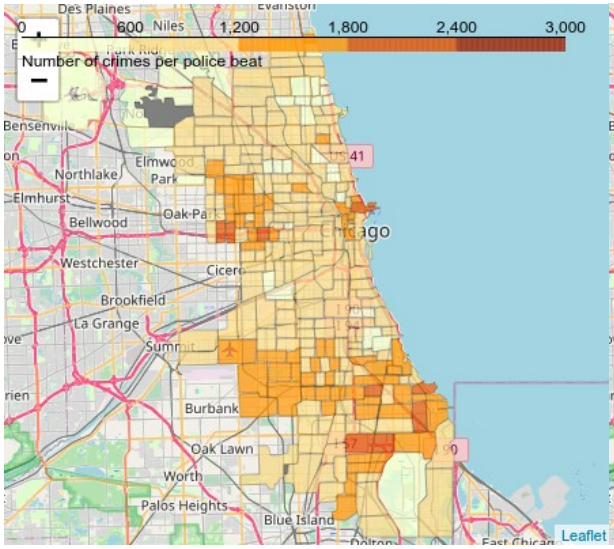
We can see that districts 21 and 31 are pretty abnormal, it's like there's almost nothing going bad there. However after we checked with data we found out as of today March 5 2022, there's only 4 cases recorded for district 21, and 221 cases related to district 31. We went on curiously checking out what is really going on in that area and surprisingly enough all the 4 cases happened in exactly the same block:

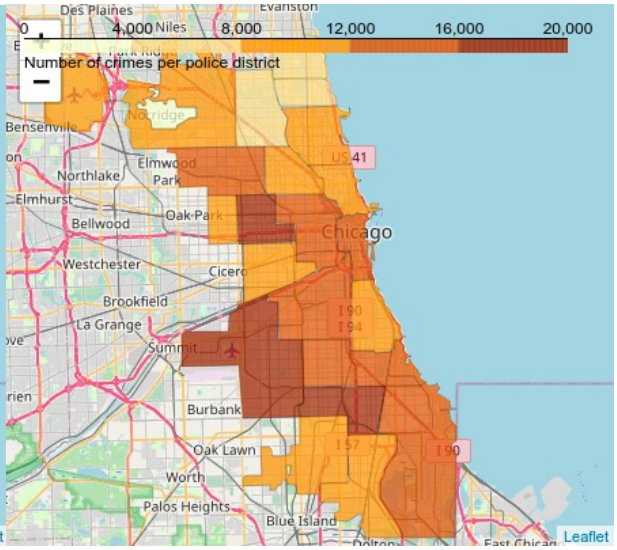
	Case Number	Date	Block	IUCR	Primary Type	Description	Location Description	Arrest	Domestic	Beat	District	Ward	Community Area	FBI Code
2289534	HJ506703	2003-07-20 00:05:00	031XX S INDIANA AVE	0890	THEFT	FROM BUILDING	OTHER	False	False	2112	21	3	35	06
2527992	HK190744	2004-02-18 18:15:00	031XX S INDIANA AVE	0420	BATTERY	AGGRAVATED:KNIFE/CUTTING INSTR	SIDEWALK	False	False	2112	21	3	35	04B
2722815	HK473636	2004-07-04 15:45:48	031XX S INDIANA AVE	2024	NARCOTICS	POSS: HEROIN(WHITE)	STREET	True	False	2112	21	3	35	18
2754437	HK530382	2004-08-01 20:05:00	031XX S INDIANA AVE	0486	BATTERY	DOMESTIC BATTERY SIMPLE	VEHICLE NON-COMMERCIAL	True	True	2112	21	3	35	08B

According to the dataset description, due to privacy issues, the exact locations are intentionally relocated, but not more than a block away from the real location, this is area where looks like a heaven, checking the data:



Other geographical graphs come in handy when dealing with some kind of area data like DISTRICT, WARD, BEAT, and COMMUNITY AREA. All the following graphs are drawn to satisfy such curiosities.





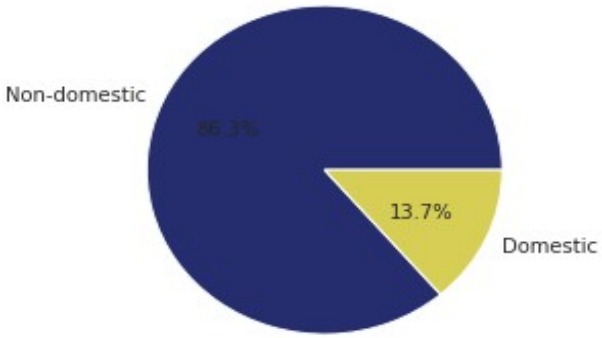
Challenge #5 Working with Choropleths and GeoJSON

Other descriptive analytics can be made to further detect pattern and as we saw our perspective gets more and more wider and detailed but this could be beyond the scope of this small demonstration, and we stop here and this concludes this part. We next delve into preprocessing which can itself involve huge amounts of workloads.

2.4. Data Preparation

And here comes the dirty part boys, just get ready! Well, not that much apparently, because the data itself is nicely recorded, gets updated at a regular schedule, and few missing values was shown in our investigations, so I'll leave that apart to you, if you wish you can easily mess around with it at your command, add a few pipelines would be much appreciated and welcomed. So I just quickly and briefly some notable discoveries so far.

The first problem we had with the data was a slight imbalanced-ness, if I may address that as that term, as there's shown clearly below:



1:6 is considered to be a small favor in the sampling to these kinds of crimes, thus we might be considering three ways to deal with that:

Either create a baseline model with the probability of guessing the dominant class, 86.3% of the times, and these models usually have some outstanding score, regarding the metrics being evaluated. Hence beating these models could be both challenging but also assuring that you're creating something better than mere guessing for sure!

The other two ways are penalizing the model on either of the classes, or using a statistical method known as undersampling, we might create k folds of data

(understanding how many fold(k: do we agree that k is a natural number?) would be needed, can further be examined, again using statistical methods)

Challenge #6 Looks like Feature Engineering never ceased to be a challenge

As the mere feature engineering, we created haversine distance metric, created using (lat, lon) points on the map. It turned out to be more in relationship(Pearson's r speaking) with the target variable(Domestic) than many other variables.

Challenge #7 Time Series Analysis

2.5. Data Modeling

This is by far the most exciting part of the field, where you relax back, try out a few models, see how the machine reacts to it, check its sanity and start to get a feel about where to go next.

Challenge #8 Time Series Forecasting

Challenge #9 Ignoring linear models all together?

Although linear models at the beginning showed little performance compared to its boosting methods counterparts, we must confess that we did little to nothing to make this data more amenable to the linear family of models.

Challenge #10 Running boosting algorithms to GPU

Challenge #11 Hyperparameter Tuning with optuna

We carefully used hyperparameter tuning using optuna, but mostly to our boosting models which showed to be more powerful in this area.

Challenge #12 Running boosting algorithms to Spark

2.6. Model Evaluation and Interpretation

Model evaluation is directly affected by the business needs and objectives. In our case, let's assume we have finalized a model, but how are we sure that this model is optical for our needs. There's a trade-off always lurking around somewhere in our modeling waiting to be revealed, and this project is no different in this manner. We chose **Recall** as

the best indicator of our models' performance since it shows the importance of the lives it could be saved due to just in time medical attention as well as extra cautious police enforcement.

Challenge #13 Which metric to choose? Type I vs. Type II errors?

2.7. Conclusion and Communicating Results

Our models' tested for this project, come to this final report which shows some other important metrics as well as the time it took to train the model:

	Performance			
	Recall	F1 Score	AUC Score	Training Time
Stochastic Gradient Classifier	0.65	0.71	0.69	00:00:05
Logistic Regression	0.66	0.70	0.69	00:00:23
eXtreme Gradient Boosting	0.79	0.83	0.82	00:13:06
Light Gradient Boosting	0.80	0.84	0.83	00:01:08
Catboost with GPU	0.85	0.82	0.81	00:00:11
XGB with GPU 5-fold Validation	0.88	0.83	0.82	00:02:50

Challenge #14 Making a dynamic table showing results right into the notebook

As you might have imagined already, there's lots of considerations, and much to learn from this dataset alone. We learned that ensemble methods, performed better than their linear counterparts, still it should be clarified that we didn't get into much of a detail of feature engineering data enough to make them more amenable to a linear model.

There's a lot of openness to the problem as we think about it more and more, sculpting data enough, considering external data sources, trying out other sampling method, trying to do some text mining, might as well get ourselves busy with time-series forecasting and using that information for additional feature to the data at hand.

The End

References:

https://chicago.suntimes.com/2021/10/29/22751518/downtown-chicago-loop-shootings-murder-river-north, *Retrieved on Feb 23 2022*