**DS8003 Management of Big Data and Tools F2021 Final Project**

**Project Documentation**

**Chicago Crime Data Analysis**

**Group Members: Nikhil Sunil, Neha Sunil, Swayami Bera**

1. **Problem Definition**

Chicago has been in the news a lot in the last few years especially due to the recent upsurge of crimes ranging from offenses like theft, assault, property damage to other violent crimes. To truly understand the true nature of this violence, we have decided to dig deeper into the crime-dataset available from the City of Chicago.

Our aim is to analyze patterns in crime and unearth insights into several questions that ensure that the police forces of Chicago are rightly distributed across the city to ensure safety and security in certain zones of the city prone to criminal activity.

To achieve our goal, we have uncovered the following insights from our analysis:

**Common Crimes in Chicago:**

* A graph describing the most prevalent offenses in Chicago over the years.
* Which crimes are most frequently committed?

**Arrests and the City of Chicago**

* Distribution of arrests and the relationship between arrest and crime rates over the years.

**Crime vs Location**

* Are there any trends in locations where the crimes are being committed?
* Distribution of different crimes over the top 10 crime affected locations.

**Crime vs Police District**

* Analysis of the top 2 crimes in the different police districts of the city.

**Crime vs Time**

* Distribution of crimes over the years
* Visualizing crime patterns over 24 hours
* Crime distribution of each crime during each hour

1. **Data Description**

**i. Attributes description**

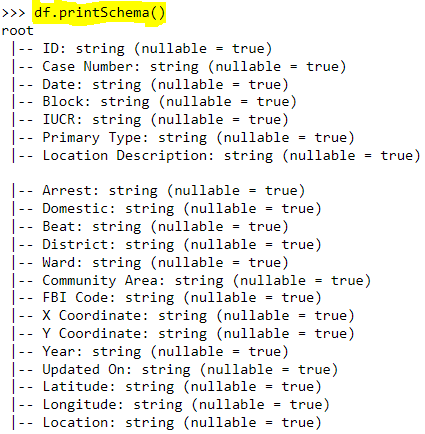
The Chicago Crime Dataset gives the details of all the crimes reported to the Chicago Police Department in the time period of 2001-2021.

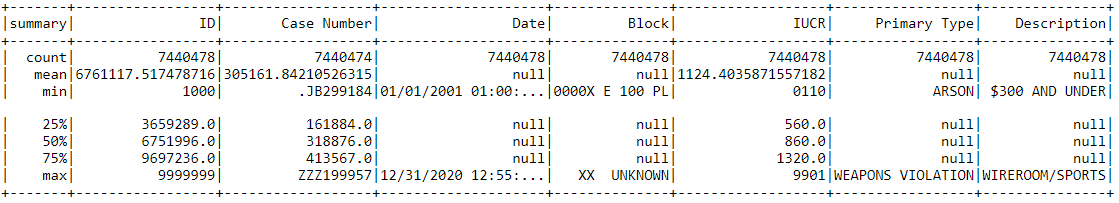
Open-source link to download the dataset:<https://data.cityofchicago.org/Public-Safety/Crimes-2001-to-Present/ijzp-q8t2/data>

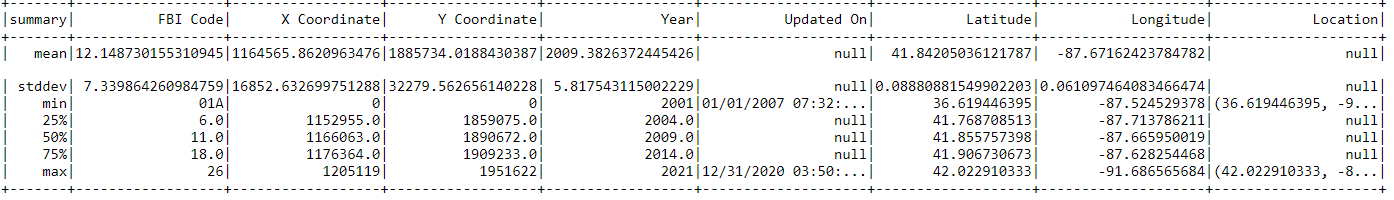
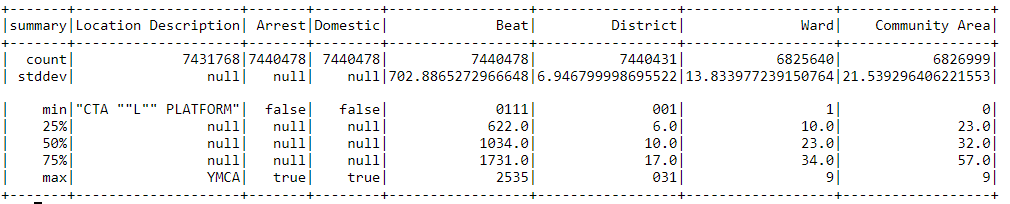
Data attributes that were used for our analysis:

1. **Date**: Date and time when the incident occurred.
2. **Primary Type**: For example, Weapons violation, Theft, Battery, Robbery, Criminal Damage, Assault, etc.
3. **Location Description**: For example, Residence, Street, Parking Lot/Garage, Apartment, etc.
4. **Arrest**: True if an arrest was made and False if an arrest wasn’t made.
5. **District**: District number where the incident occurred.
6. **Year**: Year when the incident occurred.

**ii. Statistics of the data**

****

****

****

There are a total of 7440478 entries in the Chicago Crime Dataset.

1. **Work Distribution**

The team members performed their roles accordingly. Nikhil Sunil used **HDFS** and **SPARK SQL** to query on the dataframes to get clear and incisive data. Neha Sunil on the other hand, used **Pyspark RDD, Spark SQL and visualization in python** to identify the trends and patterns in the data to come up with insights. Swayami Bera worked with **HDFS and Hive** to get data in hive tables and showcased her results by **visualizing them in python** as well.

1. **Solution Description**
2. **Tools used**

To carry out our analysis, **we have used the following tools to gather insights:**

1. Hadoop HDFS
2. Spark RDD
3. Spark SQL
4. Hive QL
5. Python for Visualization
6. **How the tools were used**

* Hadoop HDFS for storing data from the crime dataset.
* Spark RDD operations for preprocessing and transforming data.
* Spark SQL for querying data from the data set.
* Hive QL for querying data from Hive tables.
* Visualizing the data using python to present the patterns and trends gathered from our findings.

1. **Why these tools were chosen**

Hadoop HDFS was used to store the dataset. Our dataset consisted of 7440478 rows and 22 columns of data, and Hadoop was the perfect tool to store such a massive amount of data. We incorporated Spark and Hive to query the data stored in HDFS and gathered insights accordingly. Initially, both Spark and Hive seemed to have similar functions, but in order to get a first-hand understanding of how these two tools work, we used both to gather insights. We noticed that while it took over 3 minutes for each table creation in Hive, it was faster to do the same task in Spark. Moreover, working with PySpark enabled us to make use of python functionalities which was something we missed when using Hive. PySpark required a lot less data preprocessing on our end. For example, when some column names included spaces, we were able to use Pyspark queries on the dataframe as it is, but Hive posed an obstacle there. We could only move forward after renaming the columns so that it included no spaces. On the other hand, we felt that HiveQL was a lot more readable and simpler to interpret as compared to SparkSQL.

1. **Code snippets**
2. Download the dataset Crimes\_-\_2001\_to\_Present.csv to your local machine.
3. Use the following commands on Hadoop to create a directory ‘lab’, change ownership and upload the dataset to Hadoop.

**hadoop fs -mkdir /user/root/lab**

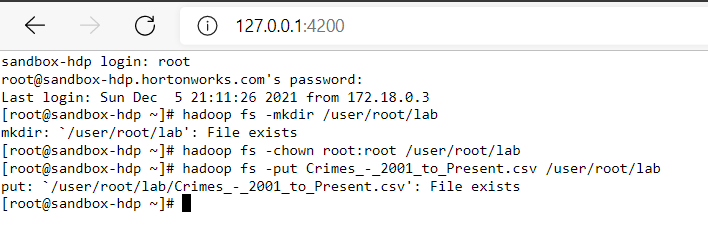
**hadoop fs -chown root:root /user/root/lab**

**hadoop fs -put Crimes\_-\_2001\_to\_Present.csv /user/root/lab**

1. Upload the dataset to the HDP sandbox via Filezilla.

-Drop the file from the left-hand side directory to /root/lab

1. Login to the sandbox with username: root and password. Change the directory to /root/lab.
2. Copy the dataset from /root/lab to HDFS /user/root/lab

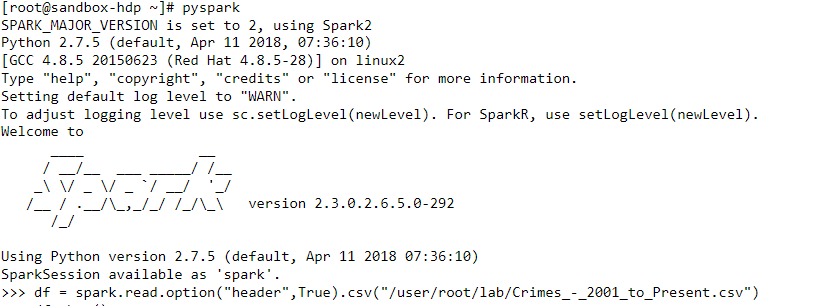
****

1. Next, we read the csv file Crimes\_-\_2001\_to\_Present.csv from HDFS using Spark

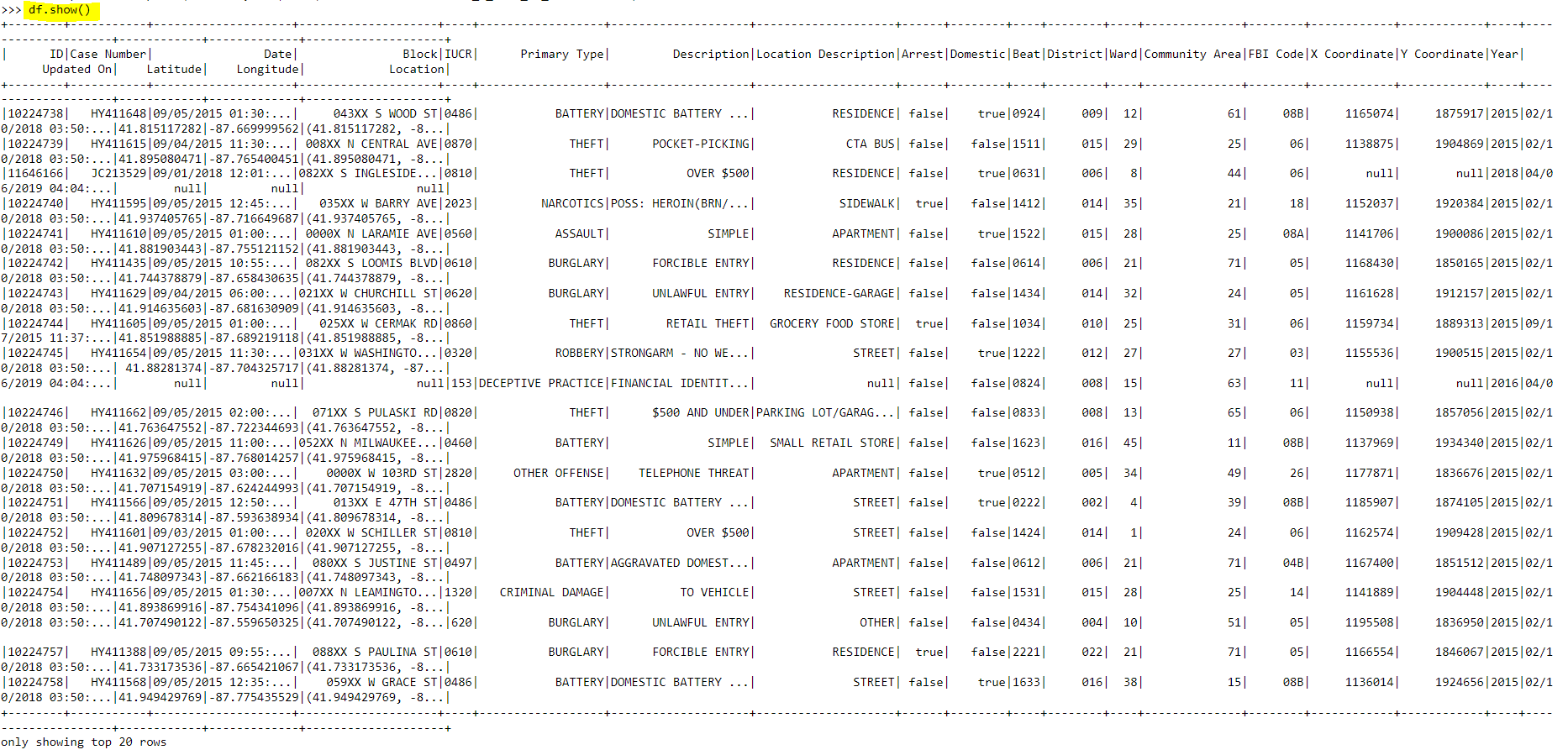
Use the following commands:

**pyspark #PySpark shell links the python API to the spark core and initializes the spark context**

**df=spark.read.option("header",True).csv("/user/root/lab/Crimes\_-\_2001\_to\_Present.csv") #this command reads the csv file and stores it in a dataframe df**

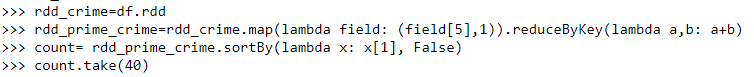
****

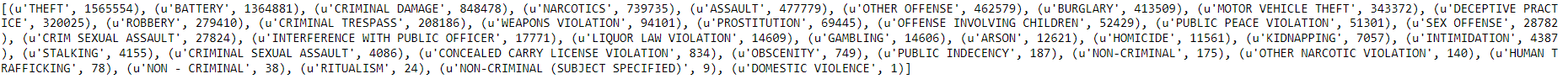
**df.show() #this command displays the top 20 rows of the dataframe df**

****

1. **Common Crimes in Chicago:**

Here we use Spark RDD operations to obtain the common crimes in Chicago and their count over the years.





**#the command rdd\_crime=df.rdd first converts the dataframe df to an rdd of strings using the .rdd function. This is stored in an rdd named rdd\_crime. Next we combine the map and reduceByKey transformations. The map transformation gets the 5th field of the rdd namely Primary Type with a count of 1. The reduceByKey operation adds the count of all the unique fields(Primary Type). We then sort the count field in descending order and assign it to a variable count which is an RDD file. The take() operation returns the first 40 values of the rdd count.**

The following commands convert the rdd file to a dataframe. We then write the output count\_df to the path “pcf.csv” which will be the directory under which the output file will be created.

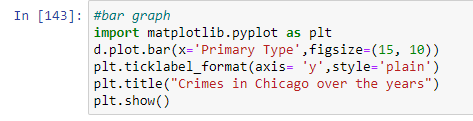


**#We write the output count\_df which contains the Primary Type crimes and their count over the years to the path “pcf.csv” which will be the directory under which the output file will be created. We do so using coalesce(1) method which partitions the data in a DataFrame. This is mainly used to reduce the number of partitions in a DataFrame.**

**CODE FOR VISUALIZATION FOR INSIGHT #1:**

**Bar Chart:**

**Fig 1a: Bar Chart describing the common crimes in Chicago over the years**

****

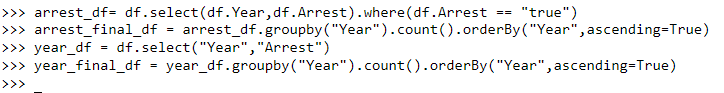
**Pie Chart:**

**Fig 1b: Pie Chart describing the most frequently committed crimes**

****

1. **Arrests and the City of Chicago**

Here we have used Spark SQL to query through the dataframe df to understand the distribution of arrests and the relationship between arrest and crime rates over the years.



**#The select query selects the columns Year and Arrest and filters them based on the condition, Arrest= ‘true’ i.e. selects the Year where Arrests were made. The results are stored in a dataframe arrest\_df. We then group the column Year from the dataframe arrest\_df by its count and sort the results in ascending order of Year. The results are stored in a dataframe arrest\_final\_df. Similarly, the columns Year and Arrest are selected from the dataframe df and the results are stored in a dataframe year\_df. We group the column Year from the dataframe year\_df by its count and sort the results in ascending order of Year. The results are stored in a dataframe year\_final\_df.**

****

****

**#We write the outputs arrest\_final\_df and year\_final\_df to the paths “arrest.csv” and “crime.csv” respectively.**

**CODE FOR VISUALIZATION FOR INSIGHT #2:**

**Bar Chart**

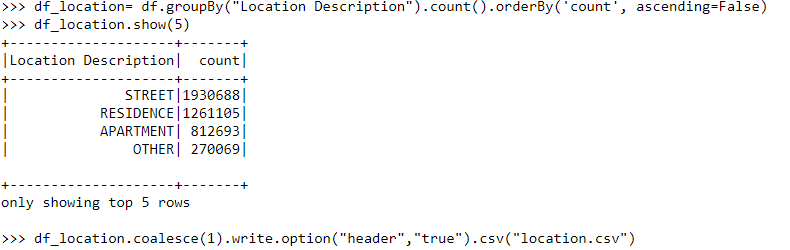
**Fig 2: Bar Chart showing the distribution of arrests and the relationship between arrest and crime rates over the years**

****

1. **Crime vs Location**

**a. Are there any trends in locations where the crimes are being committed?**

Here we use Spark SQL to query through the dataframe df to identify common trends in locations where the crimes are being committed.

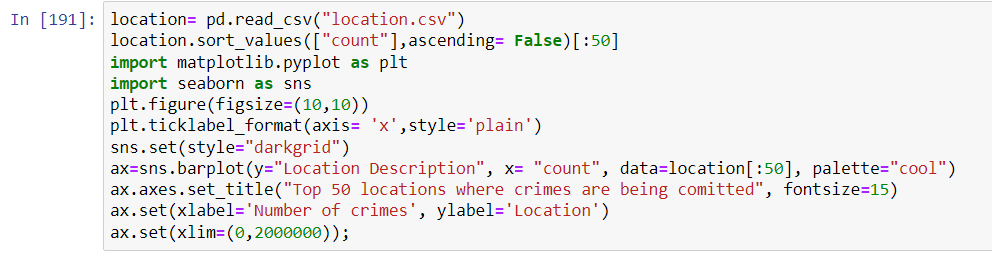


**#The above query groups the column Location Description of the dataframe df by its count and we sort the results in descending order of count using orderBy() function. The results are stored in a dataframe df\_location. We write the output df\_location which contains the Location Description of all the locations and their count over the years to the path “location.csv”**

**CODE FOR VISUALIZATION FOR INSIGHT #3a:**

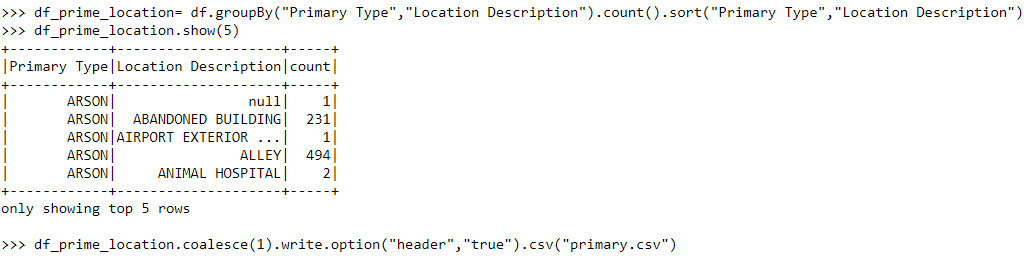
**Horizontal Bar Chart**

**Fig 3a: Bar Chart describing the top 50 locations where crimes are being committed.**

****

**b. Distribution of different crimes over the top 10 crime affected locations.**

Here we use Spark SQL to query through the dataframe df to analyze the distribution of different crimes over the top 10 crime affected locations.

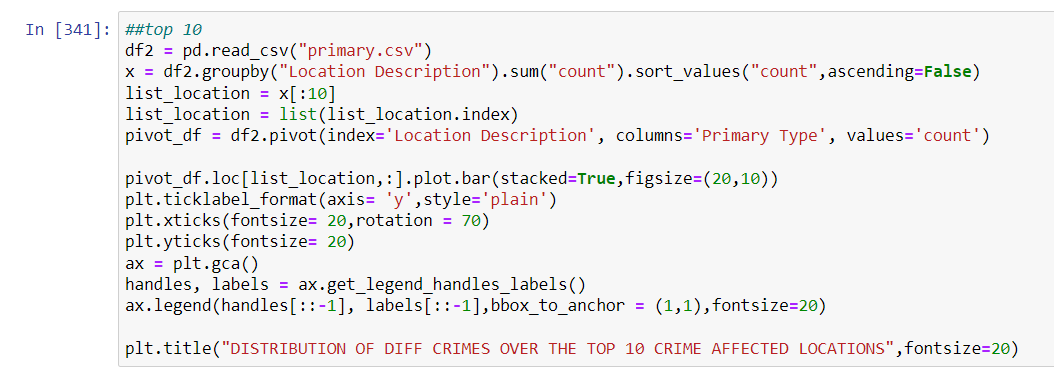


**#The above query groups the columns Primary Type and Location Description by their count and sorts the results in the order of the columns Primary Type and Location Description. We then write the output df\_prime\_location which contains the Primary Type crimes and the Location Description of all the locations where the crimes occurred and their count over the years to the path “primary.csv”**

**CODE FOR VISUALIZATION FOR INSIGHT #3b:**

**Stacked Bar Chart**

**Fig 3b: Stacked Bar Chart indicating the distribution of different crimes over the top 10 crime affected locations**

****

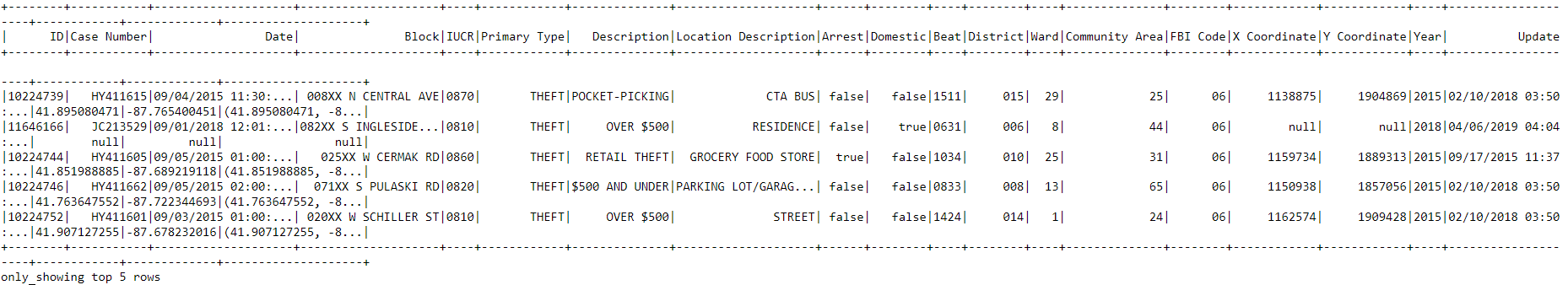
1. **Crime vs Police Districts**

Analysis of the top 2 crimes in the different police districts of the city

**a. Theft vs Police District**

Here we use Spark SQL to query through the dataframe df to analyze the rate of theft crimes over all the police districts of Chicago through the years.

****

****

****

**#The command from pyspark.sql.functions import col imports col to access the column of a dataframe. The select(\*) query selects all the records from the dataframe df and filters them based on the column value Primary Type=’THEFT’. The results are stored in a dataframe df\_theft. We then write the output df\_theft which contains all the records where Primary Type crimes are equal to ‘THEFT’ to the path “theft.csv”**

**CODE FOR VISUALIZATION FOR INSIGHT #4a:**

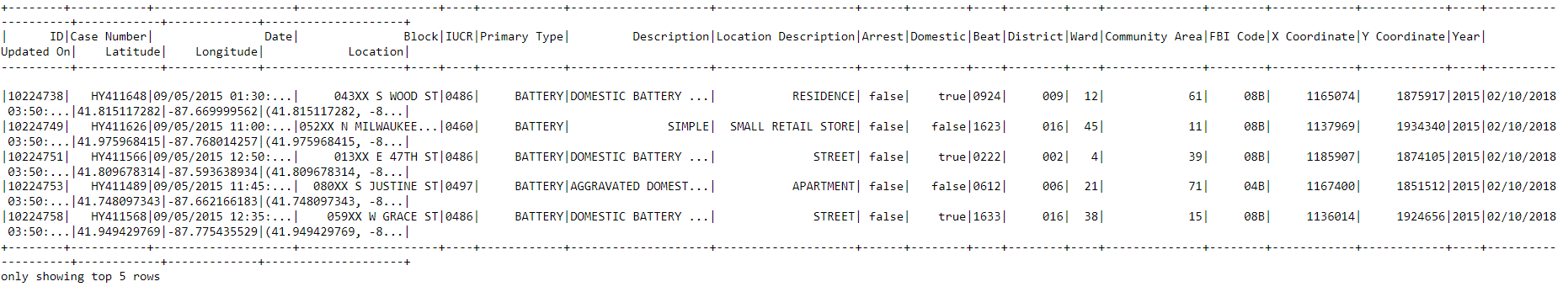
**Heatmap**

**Fig 4a: Heatmap visualizing all the Chicago Police Districts vs the number of Thefts per district from 2001 to 2021.**

****

**b. Battery vs Police District**

Here we use Spark SQL to query through the dataframe df to analyze the rate of battery related crimes over all the police districts of Chicago through the years.

****

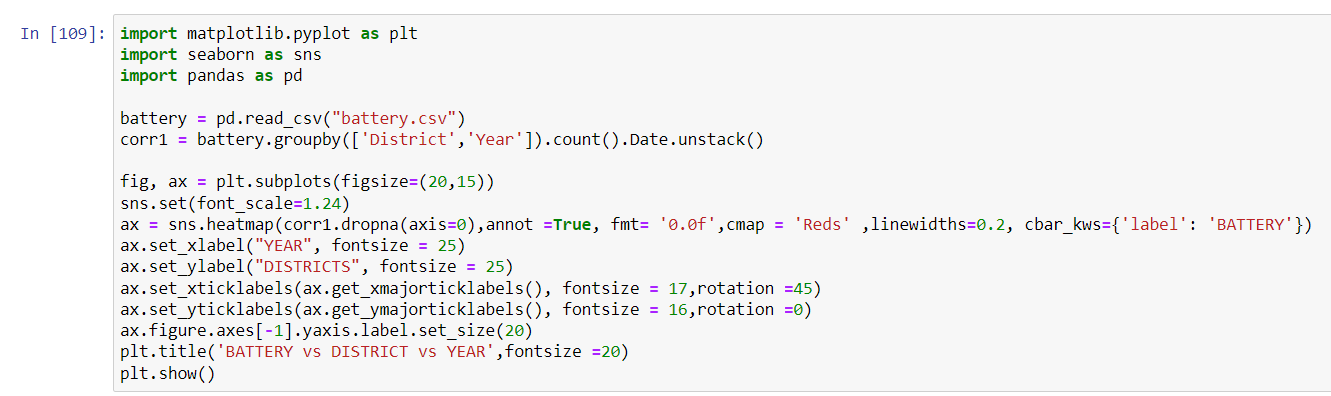
****

**#The select(\*) query selects all the records from the dataframe df and filters them based on the column value Primary Type= ‘BATTERY’. The results are stored in a dataframe df\_battery. We write the output df\_battery which contains all the records where Primary Type crimes are equal to ‘BATTERY’ to the path “battery.csv”**

**CODE FOR VISUALIZATION FOR INSIGHT #4b:**

**Heatmap**

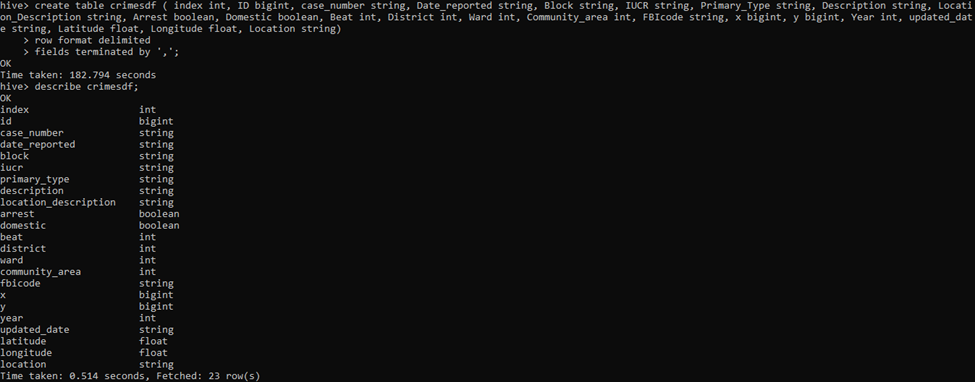
**Fig 4b: Heatmap visualizing all the Chicago Police Districts vs the number of Battery rates per district from 2001 to 2021.**

****

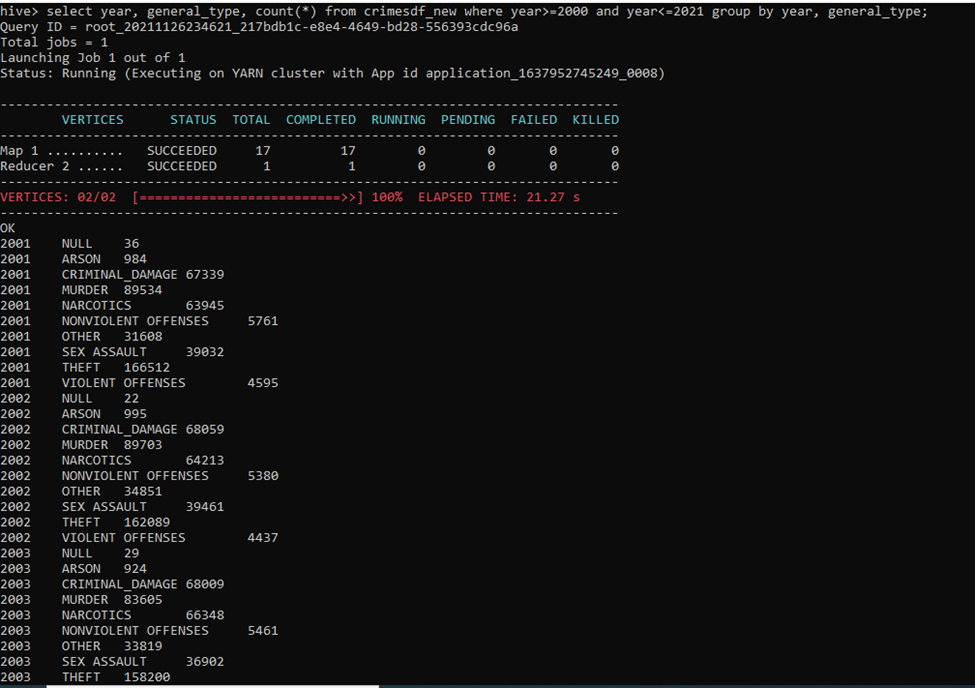
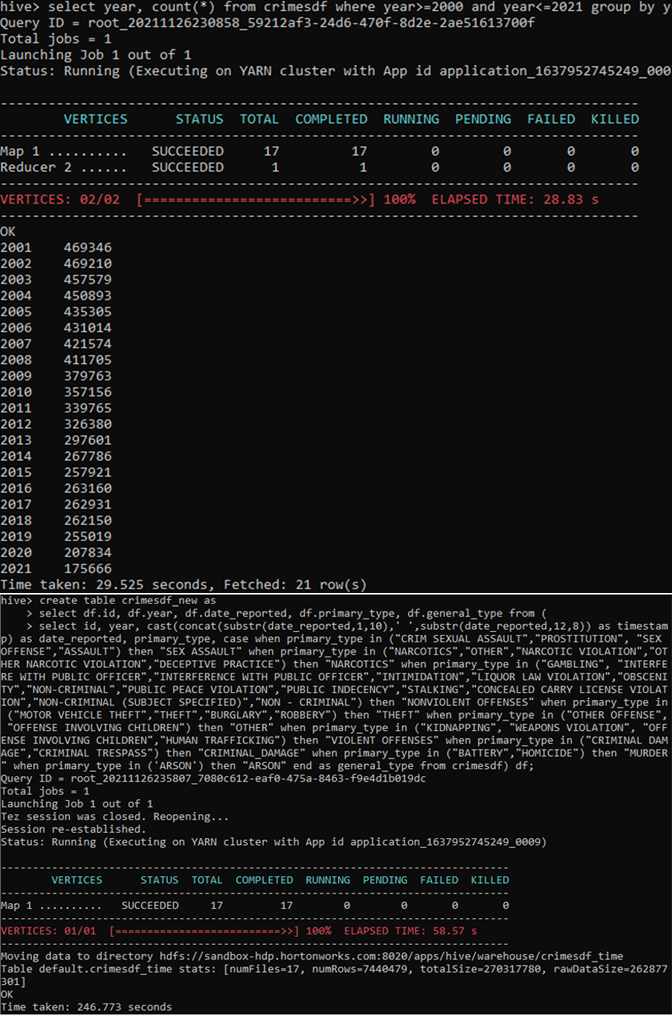
1. **Crime vs Time**

Here we have used Hive to query the dataset to understand the trends of different types of crimes over the years and the relationship between the type of crime and the time of the day.

1. **Type of crime vs year**

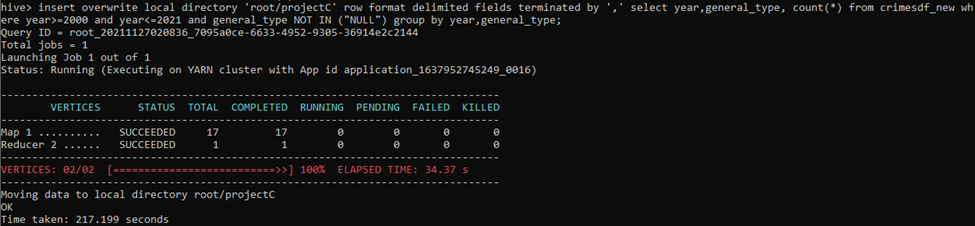
****

**#We have created a table ‘crimesdf’ that stores all the data from the HDFS. Next using the select command and the aggregate function count, we have counted the number of crimes committed when grouped by year.**

****

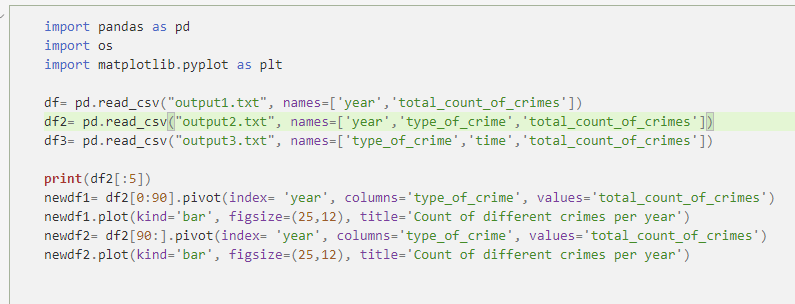
**#Here, we have first grouped the types of crimes into more general categories so that we have a better understanding of the situation at hand. And we have stored the modified data in a new table. We have grouped the contents of the modified table by year and general type to find the count of the different types of crimes committed over the years.**

**#In order to be able to visualize the data we have extracted them in the following manner.**

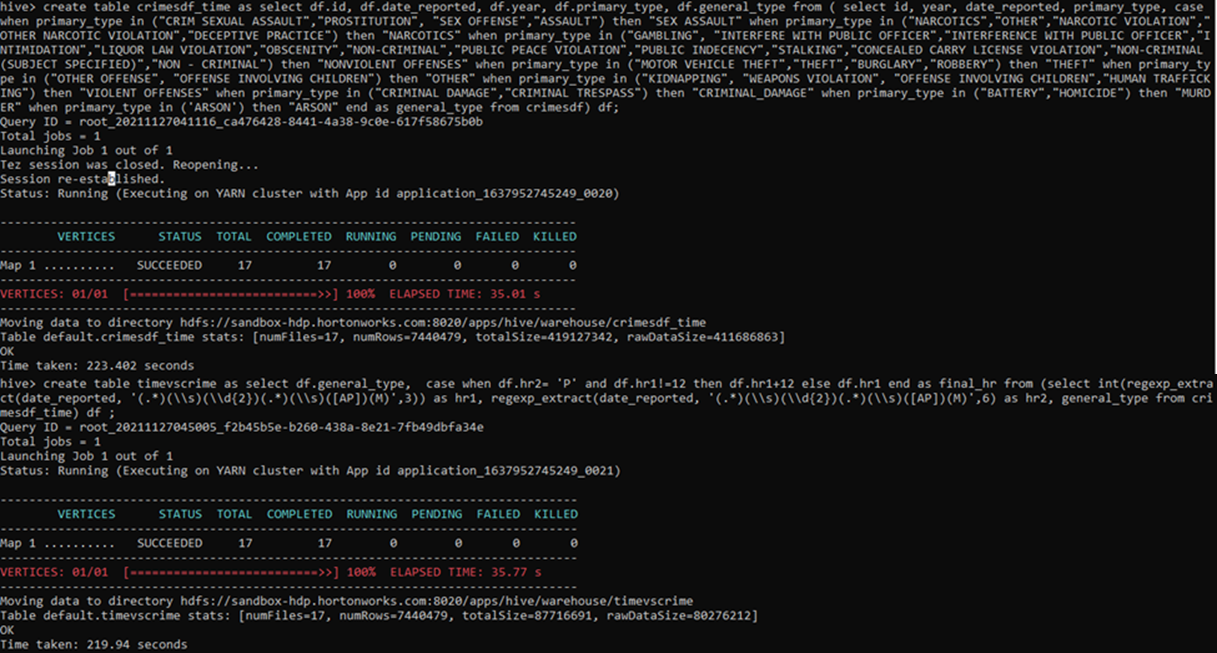
****

**CODE FOR VISUALIZATION FOR INSIGHT #5a:**

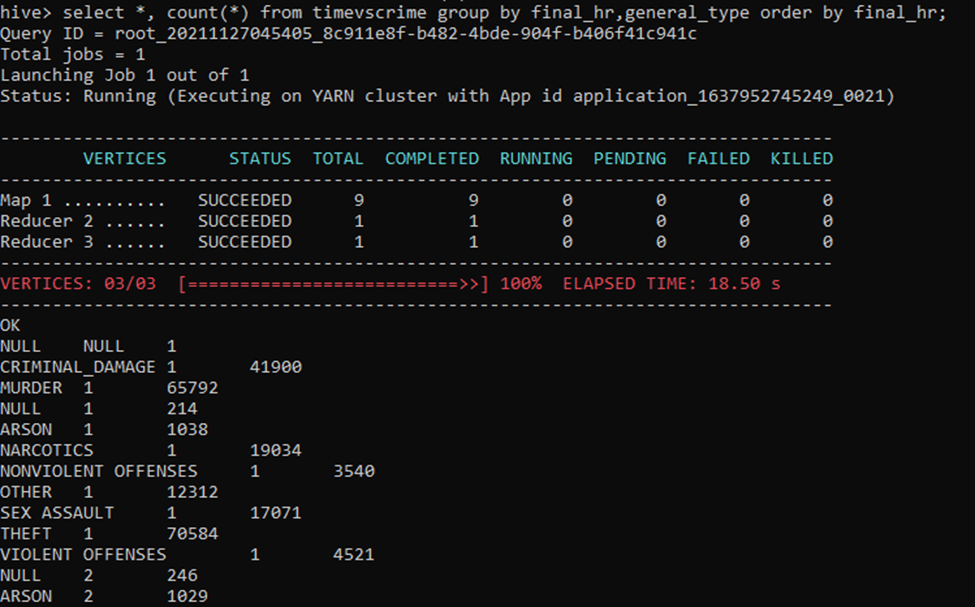
**Fig 5a and 5b: Crime vs Year**

****

1. **Type of crime vs time of the day**

****

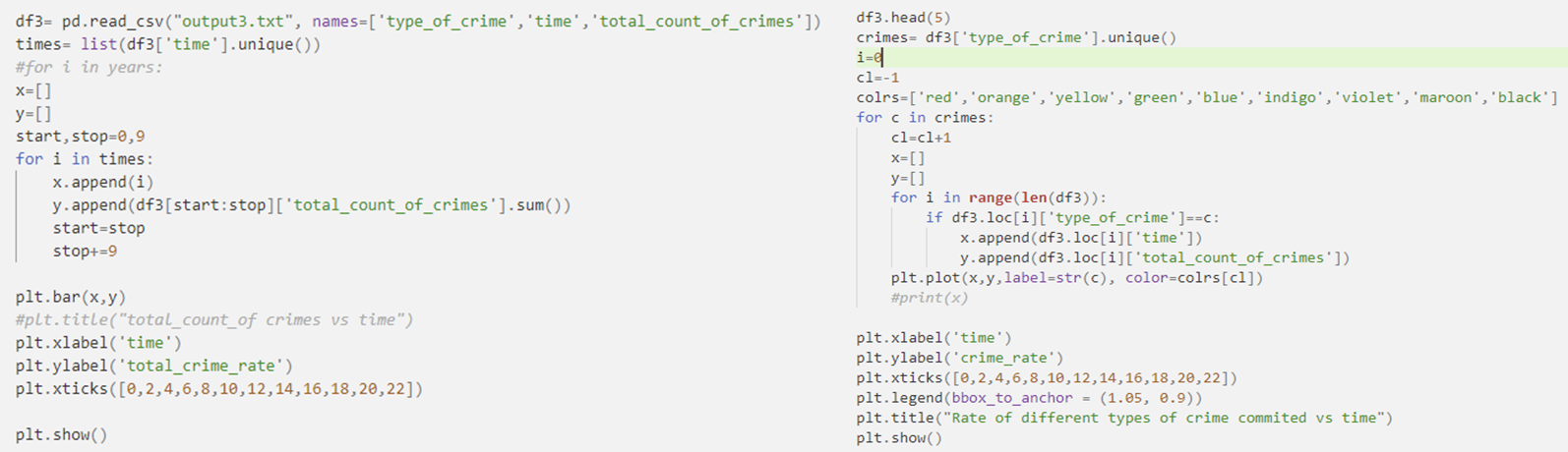
**#We created another table “crimesdf\_time” in a similar fashion as the previous one, the difference being that this includes the “date\_reported” column. The aim was to be able to extract the hour of the day from the “date\_reported” which was initially in the form of DD:MM:YYYY HH:MM:SS AM/PM.**

****

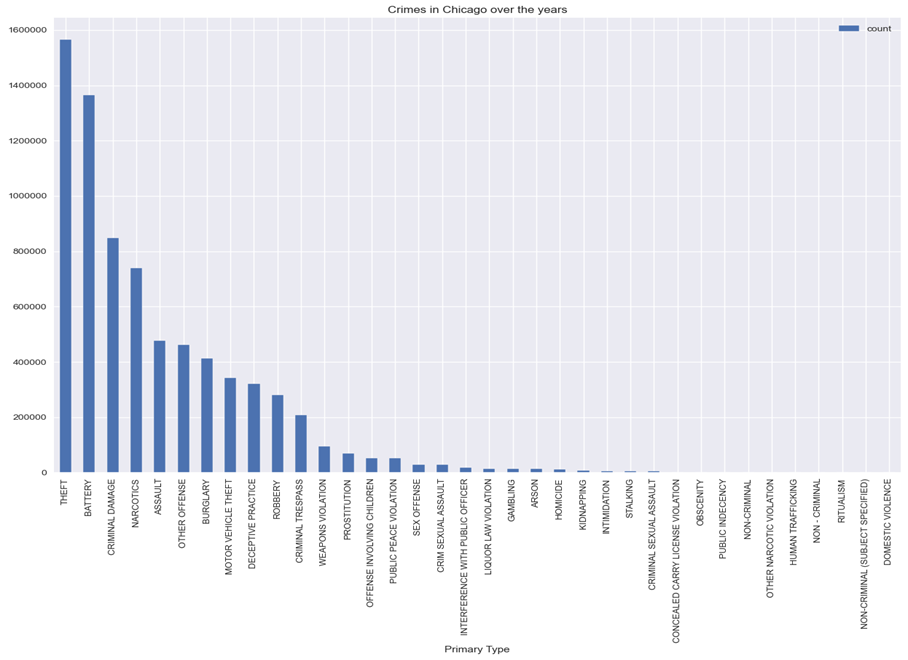
**#We have grouped the rows by hour of the day and type of crime in order to get the count of different types of crimes at different times in the day.**

**CODE FOR VISUALIZATION FOR INSIGHT #5b:**

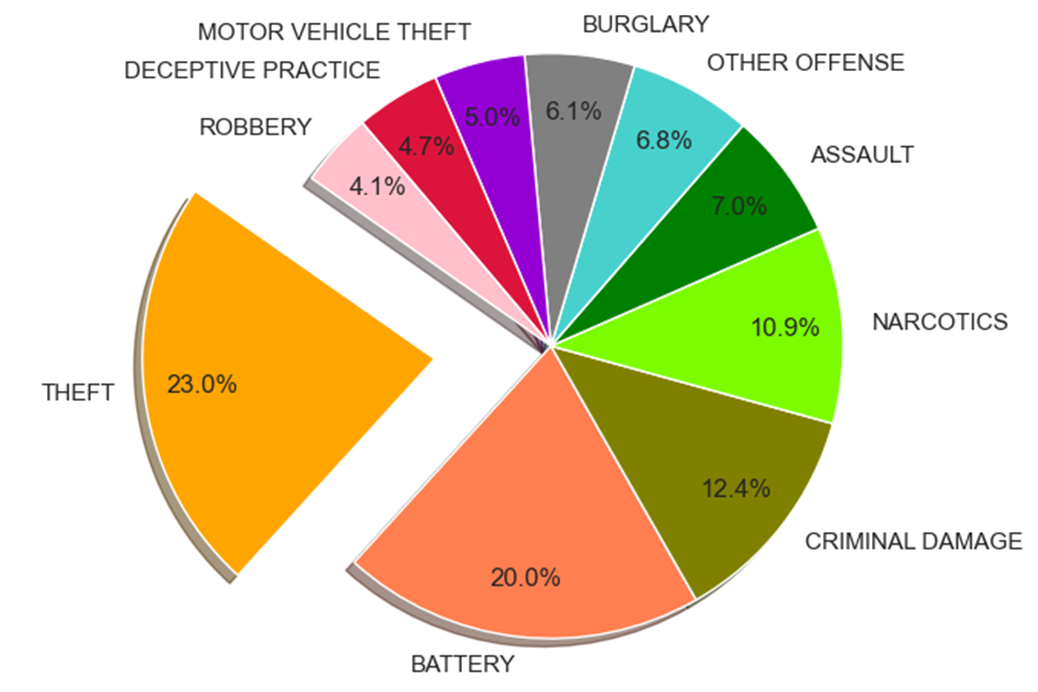
**Fig 6a and 6b: Crime vs Time of the day**

****

1. **Describing the 5 insights gathered from the data**
2. **Crimes like theft, battery (physical harm), criminal damage, narcotics and assault are far more common in Chicago and your chances of being a victim may be higher. The figures show that these crimes make up 67% of the total crimes in the city.**

****

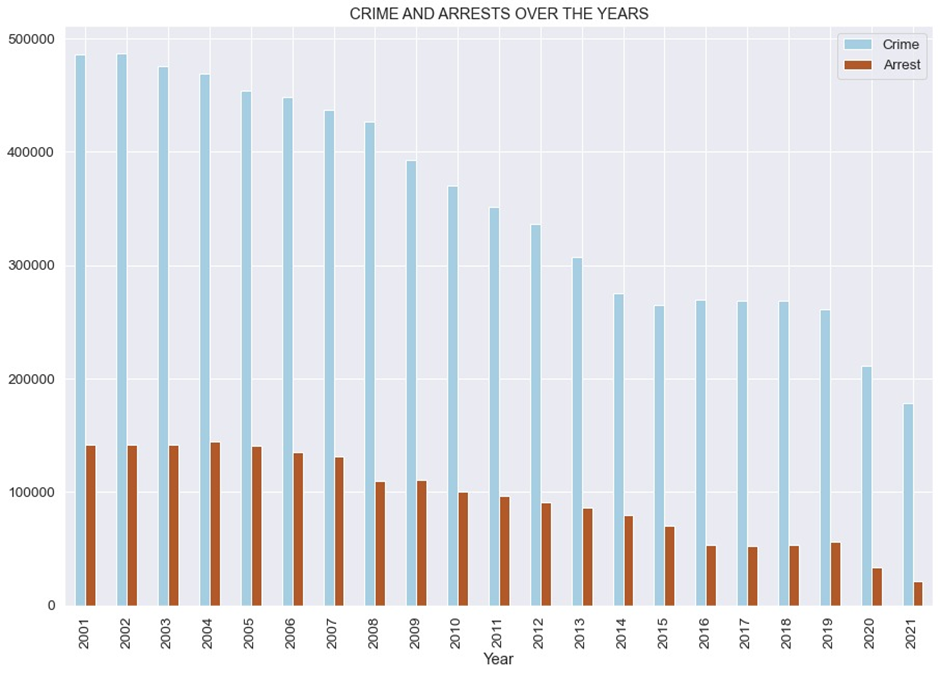
**Fig 1a: Bar Chart describing the common crimes in Chicago over the years**

****

**Fig 1b: Pie Chart describing the most frequently committed crimes**

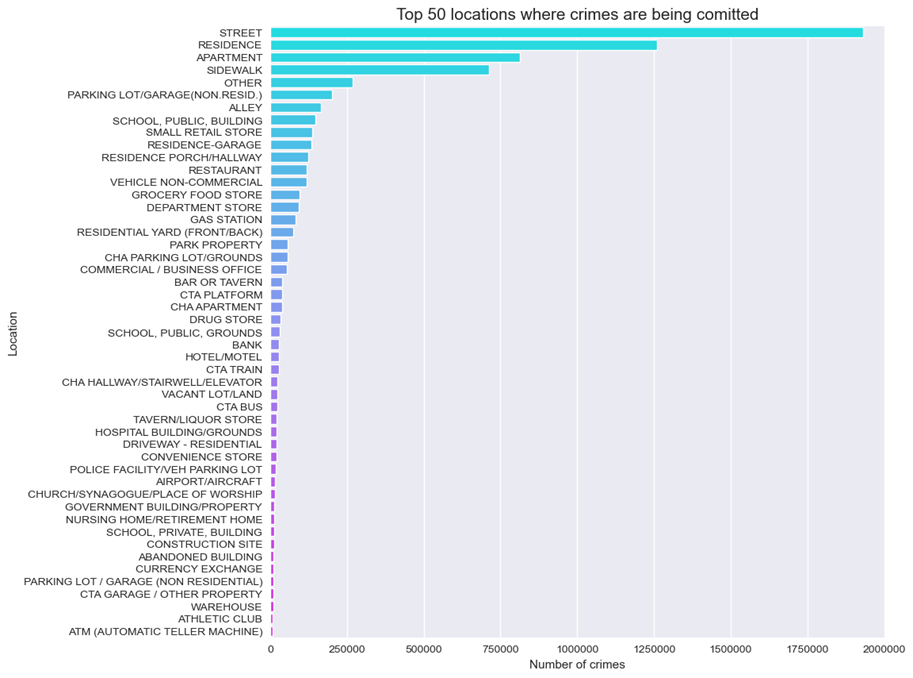
This pie chart is a statistical representation of 10 primary crime types in the city of Chicago. From the pie chart, we can see that theft is the most frequently occurring offense as it makes up 23% of the crimes. Theft, battery, criminal damage, narcotics and assault are the top 5 crimes of the city since they make up 67% of the total crimes.

1. **Although there were a lot of crimes reported in Chicago each year, the arrest rates were as low as under 50% in contrast to the number of crimes reported, indicating that the Chicago police department never found out about half of the serious crimes and in most of what they did learn of, they never made an arrest. The police tasks need to employ more effective investigation methods to create a safe living space for the people of Chicago.**

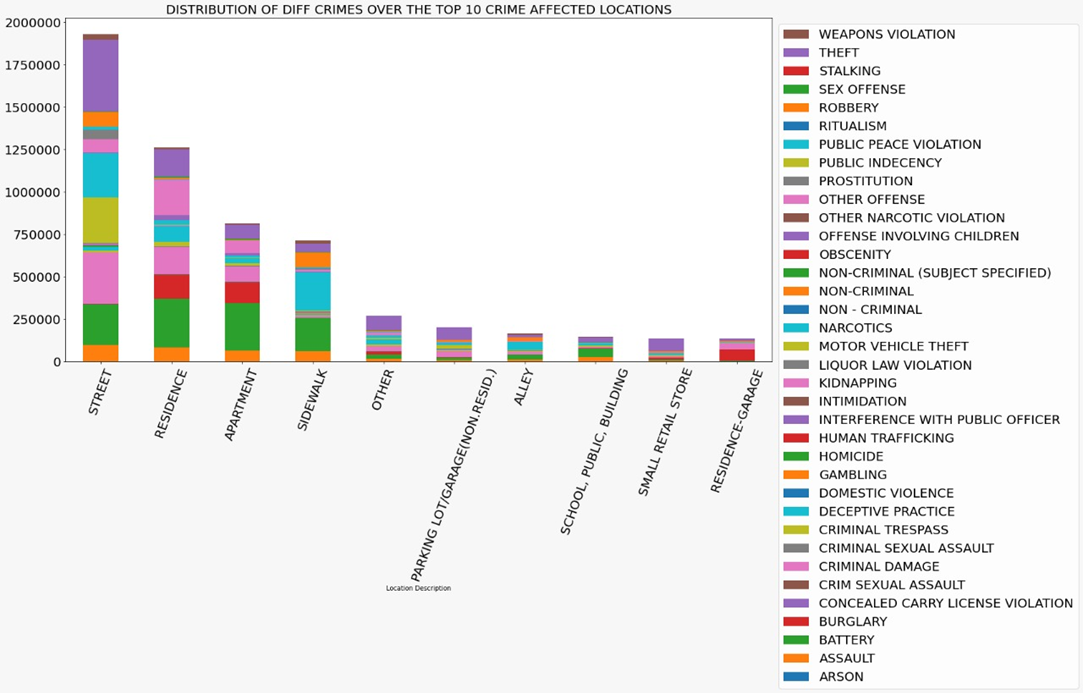
****

**Fig 2: Bar Chart showing the distribution of arrests and the relationship between arrest and crime rates over the years**

1. **Since the most common locations where crimes occur are on the streets, residences, apartments, sidewalks and parking lots, beefing up the Chicago police forces in these units can help a fair deal.**

****

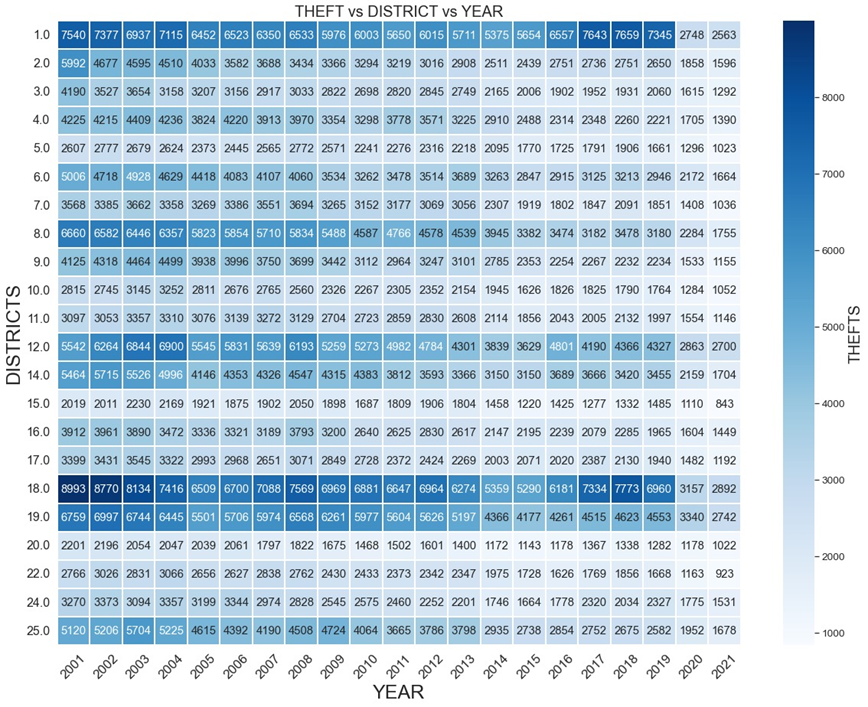
**Fig 3a: Bar Chart describing the top 50 locations where crimes are being committed.**

****

**Fig 3b: Stacked Bar Chart indicating the distribution of different crimes over the top 10 crime affected locations**

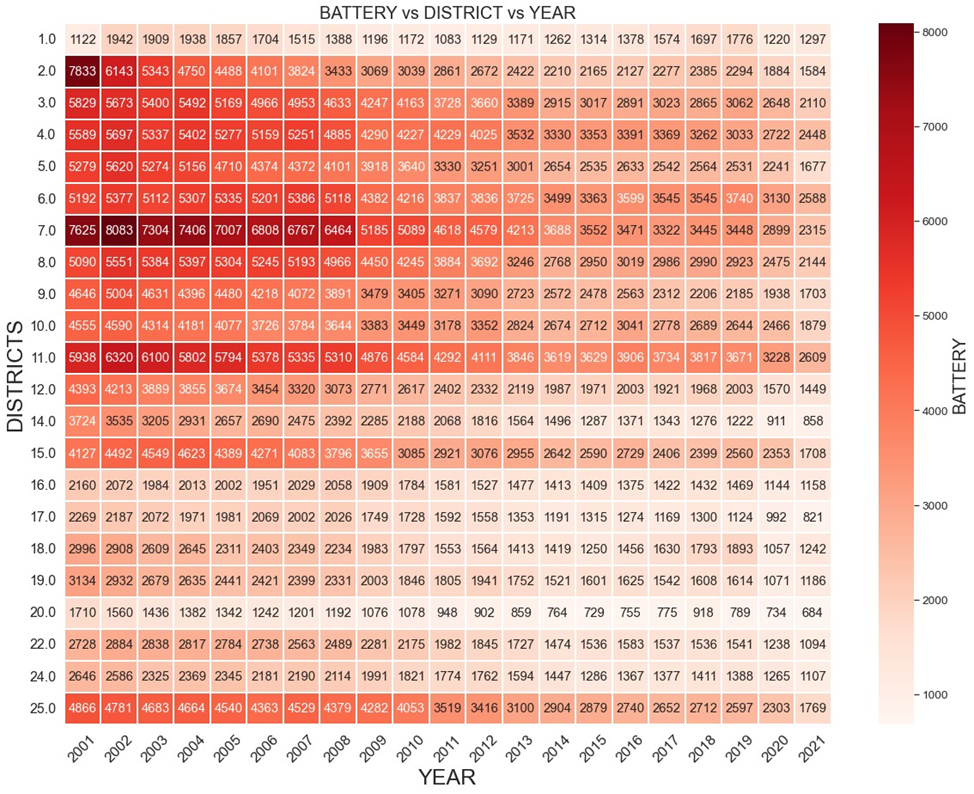
We observe that the majority of the crimes (more than 50%) occur on the streets of Chicago where it’s easier for the offenders to escape as opposed to the other locations in the city. Majority of those crimes were theft-related.

1. **The number of theft and battery crimes vary greatly by police district. District 20 had lower theft and battery rates each year for all the years in the dataset making it the safest police district in Chicago. While 2001 seemed to be the worst year for both theft and battery related crimes with over 5000 cases, the figures for 2020 and 2021 indicate a significant decline in the cases.**

****

**Fig 4a: Heatmap visualizing all the Chicago Police Districts vs the number of Thefts per district from 2001 to 2021.**

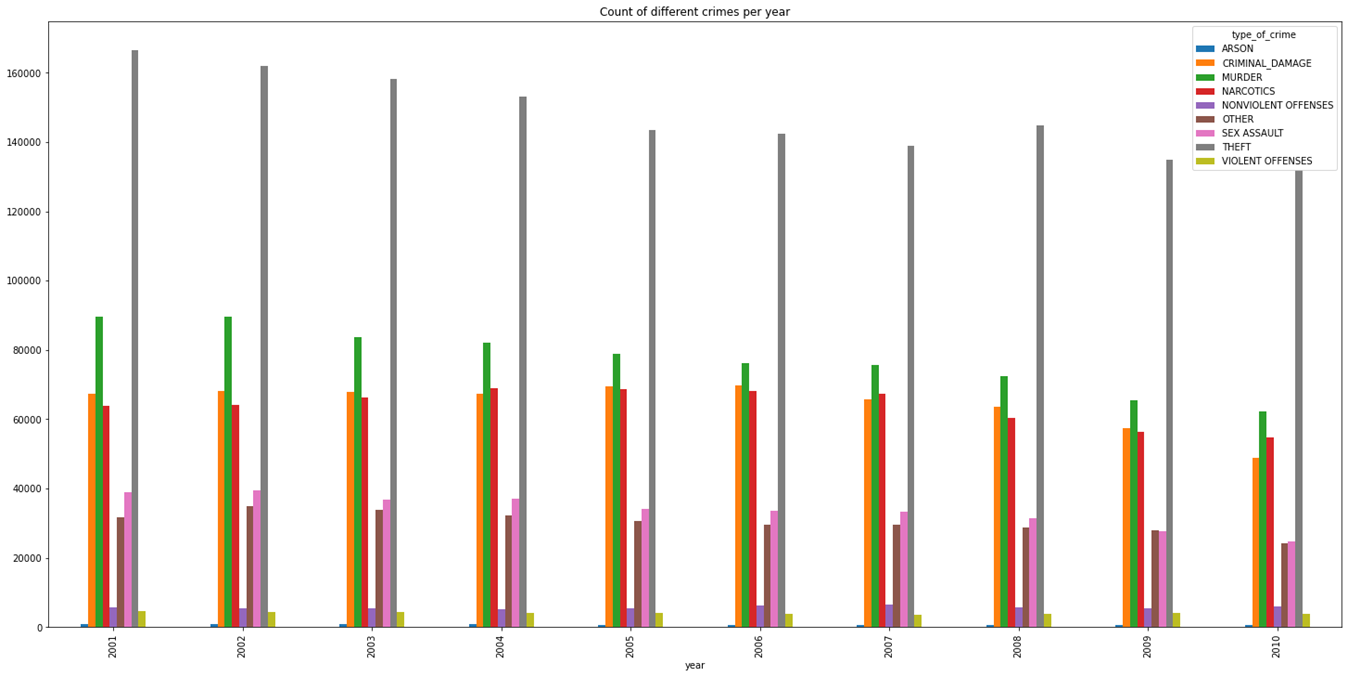
The darker regions on the map correspond to more thefts in that district in that corresponding year. Here we can see that districts 1, 18 and 19 see consistently high numbers through the 19 years in the dataset. The highest theft rate for the entire dataset was in district 18 in 2001 with 8993 thefts.

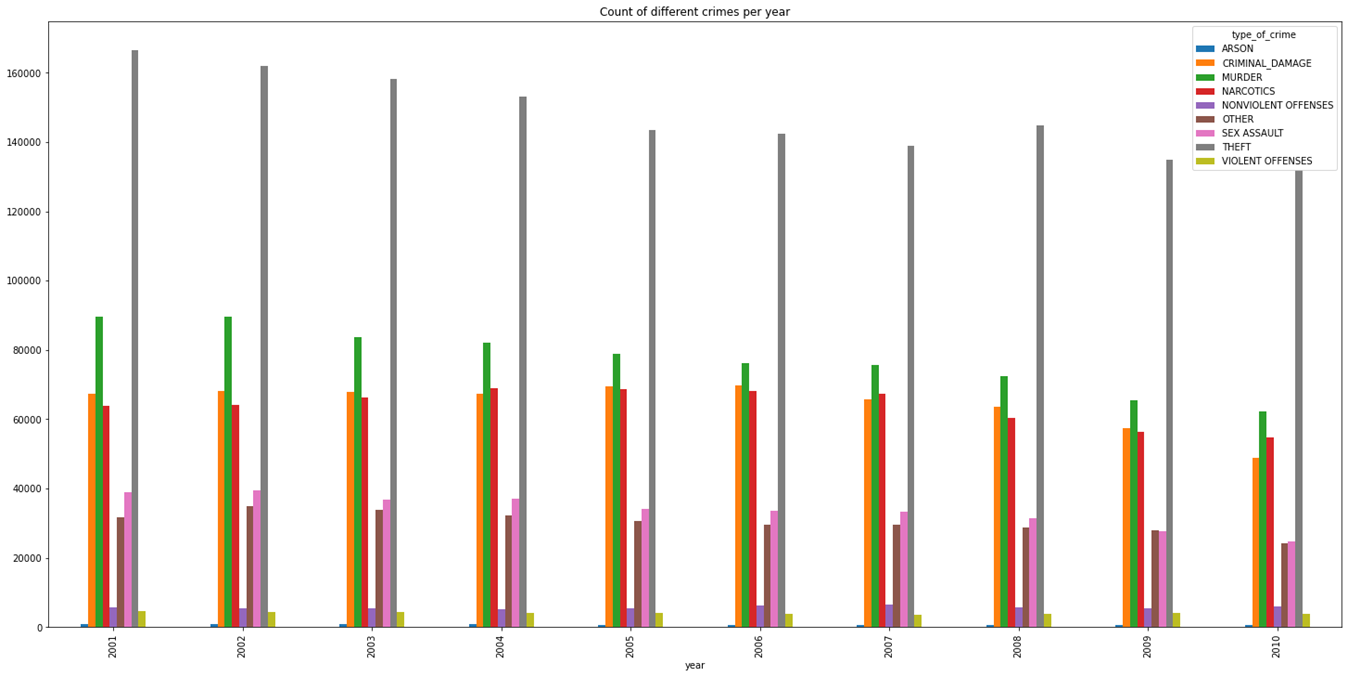


**Fig 4b: Heatmap visualizing all the Chicago Police Districts vs the number of Battery rates per district from 2001 to 2021.**

The darker regions on the map correspond to more Battery cases in that district in that corresponding year. Here we can see that districts 2, 7 and 11 saw consistently high numbers in the first 8 years from 2001 to 2008 in the dataset. The highest battery rate for the entire dataset was in district 7 in 2002 with 8083 cases.

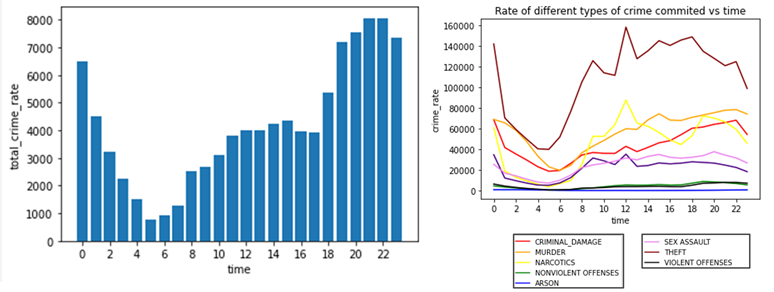
1. **On the one hand while crimes such as theft and robberies have decreased over the years, sexual assaults and narcotics related crimes are still as prevalent as they were a decade back. Special task force units could be trained to deal with such matters and to bring an end to such trafficking rings.**

****

****

**Fig 5a and 5b: Crime vs Year**

1. **Since the most vulnerable time slots determined from the graph are between 18:00-23:00 hours and 0:00-1:00 hours, more police personnel could be assigned to criminal activity prone areas at those times. We also see a spike in thefts and narcotic related crimes during noon.**

****

**Fig 6a and 6b: Crime vs Time of the day**

1. **Future Work**

In the future we would like to generate more insights based on location (latitude and longitude). The location data would give us detailed insights on crime hotspots in Chicago and further evidence about the type of crimes that occur on a regular basis in various locations. This information would benefit the Chicago police department to focus more on the red zones with high criminal activity to improve their services and to be alert to potential crimes in those zones.

1. **References**

* <https://data.cityofchicago.org/>
* <https://www.analyticsvidhya.com/blog/2016/10/using-pyspark-to-perform-transformations-and-actions-on-rdd/>
* <https://matplotlib.org/stable/api/_as_gen/matplotlib.pyplot.bar.html>
* <https://sparkbyexamples.com/apache-hive/hive-aggregate-functions-with-examples/>
* <https://spark.apache.org/docs/latest/api/python/index.html>
* <https://www.educba.com/pyspark-groupby-count/>
* <https://www.geeksforgeeks.org/how-to-add-multiple-columns-in-pyspark-dataframes/>