X dataset - Chicago crime dataset

The dataset contains various reported crimes in Chicago from 2001 till date.

Some of the important columns are

"Primary Type" - the type of crime reported. Ex: NARCOTICS, BATTERY

"Location Description" - the location where the crime is reported. Ex: STREET

"Date" - time at which the incident happened

We are considering the 67 days data from 20th Mar, 2020. As Chicago was under lockdown for 67 days from 20th Mar, 2020 and hence this might affect the crimes.

We group by just date part of the "Date" and get the data filtered for "Primary Type" as BATTERY and "Location Description" as STREET and "Primary Type" as NARCOTICS. We use these three data sets in our three inferences respectively.

We correlate the above data sets with the Illinoi related confirmed cases, deaths and confirmed cases from US-all data set respectively. Since the data is cumulative we consider the difference between successive days to get the data per day.

In [1]:

```
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
from datetime import datetime
pd.set_option("display.precision", 12)
```

In [2]:

```
from google.colab import drive
drive.mount('/content/gdrive')
```

```
Mounted at /content/gdrive
```

In [3]:

```
%cd '/content/gdrive/My Drive/Probability Project/'
```

```
/content/gdrive/My Drive/Probability Project
```

from scipy.stats import poisson, binom, geom

In [4]:

```
chicago_crime_data = pd.read_csv('chicago_crimes.csv')
```

In [ ]:

```
print(chicago_crime_data)
```

```
               ID Case Number                    Date                   Block  \
0        10224738    HY411648  09/05/2015 01:30:00 PM         043XX S WOOD ST
1        10224739    HY411615  09/04/2015 11:30:00 AM     008XX N CENTRAL AVE
2        11646166    JC213529  09/01/2018 12:01:00 AM   082XX S INGLESIDE AVE
3        10224740    HY411595  09/05/2015 12:45:00 PM       035XX W BARRY AVE
4        10224741    HY411610  09/05/2015 01:00:00 PM     0000X N LARAMIE AVE
...           ...         ...                     ...                     ...
7309444  12337793    JE196570  04/11/2021 12:00:00 AM     055XX S LA SALLE ST
7309445  12342718    JE202362  04/09/2021 03:38:00 PM    038XX W DIVERSEY AVE
7309446  12338430    JE196478  04/11/2021 05:23:00 PM     073XX S EMERALD AVE
7309447  12337810    JE196290  04/11/2021 12:20:00 PM        064XX S LOWE AVE
7309448  12337497    JE196220  04/11/2021 01:30:00 AM      005XX W MADISON ST
```

```
         IUCR              Primary Type              Description  \
0        0486                   BATTERY  DOMESTIC BATTERY SIMPLE
1        0870                     THEFT           POCKET-PICKING
2        0810                     THEFT                OVER $500
3        2023                 NARCOTICS    POSS: HEROIN(BRN/TAN)
4        0560                   ASSAULT                   SIMPLE
...       ...                       ...                      ...
7309444  1330          CRIMINAL TRESPASS                  TO LAND
7309445  1582  OFFENSE INVOLVING CHILDREN        CHILD PORNOGRAPHY
7309446  0486                   BATTERY  DOMESTIC BATTERY SIMPLE
7309447  0486                   BATTERY  DOMESTIC BATTERY SIMPLE
7309448  0910       MOTOR VEHICLE THEFT               AUTOMOBILE

        Location Description  Arrest  Domestic  ...  Ward  Community Area  \
0                  RESIDENCE   False      True  ...  12.0            61.0
1                    CTA BUS   False     False  ...  29.0            25.0
2                  RESIDENCE   False      True  ...   8.0            44.0
3                   SIDEWALK    True     False  ...  35.0            21.0
4                  APARTMENT   False      True  ...  28.0            25.0
...                      ...     ...       ...  ...   ...             ...
7309444            RESIDENCE   False      True  ...   3.0            68.0
7309445            APARTMENT   False     False  ...  30.0            21.0
7309446            APARTMENT   False      True  ...   6.0            68.0
7309447            APARTMENT   False      True  ...  20.0            68.0
7309448   RESIDENCE - GARAGE   False     False  ...  42.0            28.0

         FBI Code  X Coordinate Y Coordinate  Year              Updated On  \
0              08B     1165074.0    1875917.0  2015  02/10/2018 03:50:01 PM
1               06     1138875.0    1904869.0  2015  02/10/2018 03:50:01 PM
2               06           NaN          NaN  2018  04/06/2019 04:04:43 PM
3               18     1152037.0    1920384.0  2015  02/10/2018 03:50:01 PM
4              08A     1141706.0    1900086.0  2015  02/10/2018 03:50:01 PM
...            ...           ...          ...   ...                     ...
7309444         26     1176237.0    1868181.0  2021  04/18/2021 05:28:06 PM
7309445         17           NaN          NaN  2021  04/18/2021 05:28:06 PM
7309446        08B     1172569.0    1856288.0  2021  04/18/2021 05:28:06 PM
7309447        08B     1173144.0    1862203.0  2021  04/18/2021 05:28:06 PM
7309448         07     1172508.0    1900293.0  2021  04/18/2021 05:28:06 PM

             Latitude     Longitude                      Location
0         41.815117282 -87.669999562  (41.815117282, -87.669999562)
1         41.895080471 -87.765400451  (41.895080471, -87.765400451)
2                  NaN           NaN                           NaN
3         41.937405765 -87.716649687  (41.937405765, -87.716649687)
4         41.881903443 -87.755121152  (41.881903443, -87.755121152)
...                ...           ...                           ...
7309444   41.793645207 -87.629284614  (41.793645207, -87.629284614)
7309445            NaN           NaN                           NaN
7309446   41.761091088 -87.643084885  (41.761091088, -87.643084885)
7309447   41.777309867 -87.640802922  (41.777309867, -87.640802922)
7309448   41.881846294 -87.642010780   (41.881846294, -87.64201078)

[7309449 rows x 22 columns]
```

In [ ]:

```
chicago_crime_data.describe()
```

Out[ ]:

	ID	Beat	District	Ward	Community Area	X Coo
count	7.309449000000e+06	7.309449000000e+06	7.309402000000e+06	6.694614000000e+06	6.695965000000e+06	7.2372750000
mean	6.660261366367e+06	1.188130319262e+03	1.129500197143e+01	2.271669315064e+01	3.755123272000e+01	1.1645530181
std	3.291396837738e+06	7.029458108992e+02	6.946182569261e+00	1.383130767369e+01	2.153744920953e+01	1.6859710311
min	6.340000000000e+02	1.110000000000e+02	1.000000000000e+00	1.000000000000e+00	0.000000000000e+00	0.0000000000

25%	3.612466000000e+06	6.220000000000e+02	6.000000000000e+00	1.000000000000e+01	2.300000000000e+01	1.152941000000	
50%	6.649694000000e+06	1.034000000000e+03	1.000000000000e+01	2.200000000000e+01	3.200000000000e+01	1.166042000000	
75%	9.528264000000e+06	1.731000000000e+03	1.700000000000e+01	3.400000000000e+01	5.700000000000e+01	1.176362000000	
max	1.234311100000e+07	2.535000000000e+03	3.100000000000e+01	5.000000000000e+01	7.700000000000e+01	1.205119000000	

In [ ]:

```
chicago_crime_data.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 7309449 entries, 0 to 7309448
Data columns (total 22 columns):
 #   Column                Dtype  
---  ------                -----  
 0   ID                    int64  
 1   Case Number           object 
 2   Date                  object 
 3   Block                 object 
 4   IUCR                  object 
 5   Primary Type          object 
 6   Description           object 
 7   Location Description  object 
 8   Arrest                bool   
 9   Domestic              bool   
 10  Beat                  int64  
 11  District              float64
 12  Ward                  float64
 13  Community Area        float64
 14  FBI Code              object 
 15  X Coordinate          float64
 16  Y Coordinate          float64
 17  Year                  int64  
 18  Updated On            object 
 19  Latitude              float64
 20  Longitude             float64
 21  Location              object 
dtypes: bool(2), float64(7), int64(3), object(10)
memory usage: 1.1+ GB
```

In [64]:

```
df = chicago_crime_data
df["Epoch"] = df.apply(lambda row : datetime.strptime(row["Date"], "%m/%d/%Y %I:%M:%S %p
").timestamp(), axis = 1)
# df["Epoch"] = datetime.strptime(df["Date"], "%Y-%m-%d %I:%M:%S %p")
print(df)
```

```
              ID Case Number  ...                      Location         Epoch
0       10224738    HY411648  ...  (41.815117282, -87.669999562)  1441459800.0
1       10224739    HY411615  ...  (41.895080471, -87.765400451)  1441366200.0
2       11646166    JC213529  ...                            NaN  1535760060.0
3       10224740    HY411595  ...  (41.937405765, -87.716649687)  1441457100.0
4       10224741    HY411610  ...  (41.881903443, -87.755121152)  1441458000.0
...          ...         ...  ...                            ...          ...
7320036  12355178    JE216224  ...  (41.766592052, -87.621721416)  1619889960.0
7320037  12354571    JE216829  ...  (41.698116809, -87.698702789)  1619901000.0
7320038  12358852    JE221896  ...   (41.847351577, -87.71121355)  1614694380.0
7320039  12354228    JE216131  ...  (41.774146678, -87.615478975)  1619884860.0
7320040  12353909    JE215976  ...  (41.729264191, -87.551237878)  1619872200.0

[7320041 rows x 23 columns]
```

In [65]:

```
lowerDateLimit = df["Epoch"] >= datetime.strptime("01/22/2020 12:00:00 AM", "%m/%d/%Y %I:
```

```
%M:%S %p").timestamp()
```

In [66]:

```
upperDateLimit = df["Epoch"] <= datetime.strptime("03/04/2021 12:00:00 AM", "%m/%d/%Y %I:
%M:%S %p").timestamp()
```

In [67]:

```
filtered_crimes = df.loc[lowerDateLimit & upperDateLimit]
```

In [75]:

```
filtered_crimes.shape
```

Out[75]:

```
(227504, 23)
```

In [78]:

```
filtered_crimes_sorted = filtered_crimes.sort_values('Epoch')
```

In [69]:

```
filtered_crimes.sort_values('Epoch').groupby('Location Description')['ID'].nunique().sor
t_values()
```

Out[69]:

```
Location Description
CHA HALLWAY                1
FARM                       1
DRIVEWAY                   1
CHA ELEVATOR               1
CHA GROUNDS                1
                         ...
SMALL RETAIL STORE      5561
SIDEWALK               13727
APARTMENT              40789
RESIDENCE              42197
STREET                 54846
Name: ID, Length: 166, dtype: int64
```

In [70]:

```
df_filtered = filtered_crimes
```

In [71]:

```
df_filtered.to_csv('filtered_chicago_crime.csv')
```

In [5]:

```
US_confirmed = pd.read_csv('US_confirmed.csv')
```

In [7]:

```
US_deaths = pd.read_csv('US_deaths.csv')
```

In [72]:

```
filtered_chicago = pd.read_csv('filtered_chicago_crime.csv')
filtered_chicago = df_filtered
```

In [192]:

```
def pearson_correlation_coefficient(x, y):
    covariance_matrix = np.cov(x.astype(float),y.astype(float))
    pearson_statistic = covariance_matrix[0][1]/np.sqrt((covariance_matrix[0][0]*covaria
```

```
nce_matrix[1][1]))
    print("Pearson Correlation Coefficient Value is: " + "{:5.2f}".format(pearson_statis
tic))
    return pearson_statistic
```

Inference 1

Battery related crimes in Chicago and the Covid confirmed cases in the state of Illinois are inversely correlated.

As covid period enforces lockdown in the state of Illinois, we are expecting a fall in the battery related crimes outdoors, as we are also expecting a rise in the domestic violence cases than usual but we are not sure. So, we are taking null hypothesis and testing it using pearson's correlation statistic.

It will be useful as our test will help authorities determine where to invest more resources (if it turned out battery crimes are decreasing because of covid authorities and shift some of the resources on this to another issues)

Since the null Hypothesis is checking for inverse linear correlation. We take the threshold = -0.5

In [196]:

```
#Inference 1
filtered_crimes_sorted = df_filtered

filtered_crimes_sorted = filtered_crimes_sorted[(filtered_crimes_sorted['Primary Type']
== "BATTERY") & (filtered_crimes_sorted["Epoch"]> 1584676800)]
filtered_crimes_sorted["DateStr"] = filtered_crimes_sorted.apply(lambda row : datetime.f
romtimestamp(row["Epoch"]).date(), axis = 1)
data = filtered_crimes_sorted["DateStr"].value_counts(sort = False).sort_index()
data = np.array(data[58:58+67].tolist())

US_changed = US_confirmed[US_confirmed['State'] == "IL"].T.rename(index = {'State':'Date
'})
US_changed = US_changed.loc['2020-03-19':'2020-05-25']
US_changed = US_changed.diff()
US_changed = US_changed[1:]

# PEARSON CORRELATION COEFFICIENT for Battery VS Confirmed cases in the State of Illinois
pearson_coef = pearson_correlation_coefficient(data, US_changed[14].to_numpy())
```

```
Pearson Correlation Coefficient Value is:  0.39
```

```
/usr/local/lib/python3.7/dist-packages/ipykernel_launcher.py:5: SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead

See the caveats in the documentation: https://pandas.pydata.org/pandas-docs/stable/user_g
uide/indexing.html#returning-a-view-versus-a-copy
  """
```

The pearson correlation coefficient is 0.39 and is not smaller than the threshold -0.5 hence we reject the null hypothesis.

Inference 2

Number of crimes that happened on the street in Chicago VS Covid confirmed cases in the state of Illinois are inversely correlated.

As Covid rampages the streets of Chicago, people tend to find peace at their homes with their loved ones, leaving streets deserted. So, we are expecting a decrease in the crime rate on streets. We test the null hypothesis with pearson's correlation coefficient statistic.

If it turns out that the street crime actually decreased during the covid. Police and actually decrease patrol rounds and use those resources elsewhere.

Since the null Hypothesis is checking for inverse linear correlation. We take the threshold = -0.5

In [197]:

```
#Inference 2

filtered_crimes_sorted = df_filtered

filtered_crimes_sorted = filtered_crimes_sorted[(filtered_crimes_sorted['Location Description'] == "STREET") & (filtered_crimes_sorted["Epoch"]> 1584676800)]
filtered_crimes_sorted["DateStr"] = filtered_crimes_sorted.apply(lambda row : datetime.fromtimestamp(row["Epoch"]).date(), axis = 1)
data = filtered_crimes_sorted["DateStr"].value_counts(sort = False).sort_index()
data = np.array(data[58:58+67].tolist())

# PEARSON CORRELATION COEFFICIENT for Crimes reported in streets VS Confirmed cases in the State of Illinois
pearson_coef = pearson_correlation_coefficient(data, US_changed[14].to_numpy())
```

```
Pearson Correlation Coefficient Value is:  0.20
```

```
/usr/local/lib/python3.7/dist-packages/ipykernel_launcher.py:6: SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead

See the caveats in the documentation: https://pandas.pydata.org/pandas-docs/stable/user_guide/indexing.html#returning-a-view-versus-a-copy
```

The pearson correlation coefficient is 0.2 and is not smaller than the threshold -0.5 hence we accept the null hypothesis.

Inference 3

Narcotics related crimes in Chicago vs Covid related deaths in state of Illinois are linearly correlated.

Generally, more covid deaths are observed on patients having pre-existing conditions. Increase in narcotic crimes imply increase in the usage of narcotics. As people consume more narcotic substances, this weakens their immune system and are susceptible to dying because of covid. We test the null hypothesis with pearson's correlation coefficient statistic.

If it turns out that the Narcotic related crimes and covid deaths are positively correlated, then the drug control bureau can probably take some stringent norms to reduce the usage of narcotics.

Since the null Hypothesis is checking for linear correlation. We take the threshold = 0.5

In [198]:

```
#Inference 3

filtered_crimes_sorted = df_filtered

filtered_crimes_sorted = filtered_crimes_sorted[(filtered_crimes_sorted['Primary Type'] == "NARCOTICS") & (filtered_crimes_sorted["Epoch"]> 1584676800)]
filtered_crimes_sorted["DateStr"] = filtered_crimes_sorted.apply(lambda row : datetime.fromtimestamp(row["Epoch"]).date(), axis = 1)
data = filtered_crimes_sorted["DateStr"].value_counts(sort = False).sort_index()
data = np.array(data[58:58+67].tolist())

US_changed_death = US_deaths[US_deaths['State'] == "IL"].T.rename(index = {'State':'Date'})
US_changed_death = US_changed_death.loc['2020-03-19':'2020-05-25']
US_changed_death = US_changed_death.diff()
US_changed_death = US_changed_death[1:]

# PEARSON CORRELATION COEFFICIENT for Narcotics VS Deaths in the State of Illinois
pearson_coef = pearson_correlation_coefficient(data, US_changed_death[14].to_numpy())
```

```
Pearson Correlation Coefficient Value is: -0.07
```

```
/usr/local/lib/python3.7/dist-packages/ipykernel_launcher.py:6: SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame.
```

```
Try using .loc[row_indexer,col_indexer] = value instead

See the caveats in the documentation: https://pandas.pydata.org/pandas-docs/stable/user_g
uide/indexing.html#returning-a-view-versus-a-copy
```

The pearson correlation coefficient is -0.07 and is smaller than the threshold 0.5 hence we accept the null hypothesis.