IBM-DataScience-Machine_Learning-Capstone-Project

February 27, 2021

1 IBM-Capstone-Project

1.1 Description of the Problem and Discussion of the Background

1.1.1 Introduction Section:

The police of Denver, Colorado patrolling in the whole city. However, various crimes occurred in any part of the city. Due to the limited number of patrolling cars of the police department, most of the area is not covered by the police cars. Therefore, police need to know those areas, where most crimes have occurred. So that, they can cover most of the area by sending police car. The less crime area can be omitted.

To understand these, the police needs answers to several questions. The answers to those questions must be supported by data and analytics. These are their questions:

1) How can we find those areas, where the most crimes have occurred?

2) How can we find the most offensive crimes and the area?

3) How can we find the most traffic area?

1.1.2 Target Audience

The analysis would help any the police department of the city of Denver to predict the place of the crimes and types of the crimes. Moreover, the police can increase/decrease the number of patrolling cars in an area, where it's needed.

1.1.3 Data Description and Data Sources

Data used in this project is collected from several sources. A brief description of the sources of data are given below:

This dataset includes criminal offenses in the City and County of Denver for the previous five calendar years plus the current year to date. The data is based on the National Incident Based Reporting System (NIBRS) which includes all victims of person crimes and all crimes within an incident. The data is dynamic, which allows for additions, deletions and/or modifications at any time, resulting in more accurate information in the database. Due to continuous data entry, the number of records in subsequent extractions are subject to change. Crime data is updated Monday through Friday. The following image shows an some data from the crimes data:

	INCIDENT_ID	OFFENSE_ID	OFFENSE_CODE	OFFENSE_CODE_EXTENSION	OFFENSE_TYPE_ID	OFFENSE_CATEGORY_ID	FIRST_OCCURRENCE_DATE
0	20176005213	20176005213239901	2399	1	theft-bicycle	larceny	6/8/2017 1:15:00 PM
1	2016461725	2016461725549900	5499	0	traf-other	all-other-crimes	7/21/2016 6:40:00 PM
2	2017409119	2017409119549900	5499	0	traf-other	all-other-crimes	6/22/2017 5:20:00 PM
3	2016829592	2016829592110200	1102	0	sex-aslt-rape	sexual-assault	12/30/2016 11:00:00 PM
4	2017455505	2017455505544100	5441	0	traffic-accident	traffic-accident	7/10/2017 6:00:00 PM

For more info please check the following link:

https://www.denvergov.org/opendata/dataset/city-and-county-of-denver-crime

Foursquare API to explore venue types surrounding each neighborhood of the city of Denver. The query was made for the number of venues in each category within a 1000m radius around each neighborhood ("Documentation — Foursquare Developer", 2020).

[53]:
```
# importing all neccessary dependencies
import pandas as pd
pd.set_option('display.max_columns', None)
pd.set_option('display.max_rows', None)
import numpy as np
import json
import requests
import folium
from sklearn.cluster import KMeans
import matplotlib.pyplot as plt
import seaborn as sns
import re
from sklearn.cluster import KMeans
from pandas.io.json import json_normalize #for tranforming JSON file into a␣
 ↪pandas dataframe
import matplotlib.cm as cm
import matplotlib.colors as colors
```

[54]:
```
#loading crimes data of Denver City.
df=pd.read_csv('crime.csv')
df.head()
```

[54]:
```
    INCIDENT_ID         OFFENSE_ID  OFFENSE_CODE  OFFENSE_CODE_EXTENSION  \
0  20176005213  20176005213239901          2399                       1
1   2016461725   2016461725549900          5499                       0
2   2017409119   2017409119549900          5499                       0
3   2016829592   2016829592110200          1102                       0
4   2017455505   2017455505544100          5441                       0

    OFFENSE_TYPE_ID OFFENSE_CATEGORY_ID   FIRST_OCCURRENCE_DATE  \
0     theft-bicycle             larceny     6/8/2017 1:15:00 PM
1        traf-other    all-other-crimes    7/21/2016 6:40:00 PM
```

```
2        traf-other    all-other-crimes    6/22/2017 5:20:00 PM
3     sex-aslt-rape      sexual-assault  12/30/2016 11:00:00 PM
4  traffic-accident    traffic-accident    7/10/2017 6:00:00 PM

     LAST_OCCURRENCE_DATE          REPORTED_DATE             INCIDENT_ADDRESS  \
0     6/8/2017 5:15:00 PM   6/12/2017 8:44:00 AM                 1705 17TH ST
1                     NaN   7/21/2016 7:09:00 PM  N COLUMBINE ST / E 48TH AVE
2                     NaN   6/22/2017 5:20:00 PM  E COLFAX AVE / N DOWNING ST
3  12/30/2016 11:45:00 PM  12/31/2016 4:59:00 AM                          NaN
4                     NaN   7/10/2017 6:45:00 PM     E EVANS AVE / S GRAPE ST

       GEO_X      GEO_Y     GEO_LON    GEO_LAT  DISTRICT_ID  PRECINCT_ID  \
0  3140790.0  1699792.0 -104.999264  39.753669            6          612
1  3152605.0  1710822.0 -104.957009  39.783762            2          212
2  3148176.0  1694866.0 -104.973097  39.740032            6          623
3        NaN        NaN         NaN        NaN            6          611
4  3161788.0  1672521.0 -104.925197  39.678463            3          323

  NEIGHBORHOOD_ID  IS_CRIME  IS_TRAFFIC
0   union-station         1           0
1  elyria-swansea         1           0
2    capitol-hill         1           0
3     five-points         1           0
4       goldsmith         0           1
```

[55]:
```
#Removing some unnecessary columns
df=df.drop(['INCIDENT_ID', 'OFFENSE_ID', 'OFFENSE_CODE',␣
 ↪'OFFENSE_CODE_EXTENSION', 'PRECINCT_ID'], axis=1)
df.head()
```

[55]:
```
    OFFENSE_TYPE_ID OFFENSE_CATEGORY_ID   FIRST_OCCURRENCE_DATE  \
0     theft-bicycle             larceny     6/8/2017 1:15:00 PM
1        traf-other    all-other-crimes    7/21/2016 6:40:00 PM
2        traf-other    all-other-crimes    6/22/2017 5:20:00 PM
3     sex-aslt-rape      sexual-assault  12/30/2016 11:00:00 PM
4  traffic-accident    traffic-accident    7/10/2017 6:00:00 PM

     LAST_OCCURRENCE_DATE          REPORTED_DATE             INCIDENT_ADDRESS  \
0     6/8/2017 5:15:00 PM   6/12/2017 8:44:00 AM                 1705 17TH ST
1                     NaN   7/21/2016 7:09:00 PM  N COLUMBINE ST / E 48TH AVE
2                     NaN   6/22/2017 5:20:00 PM  E COLFAX AVE / N DOWNING ST
3  12/30/2016 11:45:00 PM  12/31/2016 4:59:00 AM                          NaN
4                     NaN   7/10/2017 6:45:00 PM     E EVANS AVE / S GRAPE ST

       GEO_X      GEO_Y     GEO_LON    GEO_LAT  DISTRICT_ID NEIGHBORHOOD_ID  \
0  3140790.0  1699792.0 -104.999264  39.753669            6   union-station
1  3152605.0  1710822.0 -104.957009  39.783762            2  elyria-swansea
```

```
2  3148176.0  1694866.0 -104.973097  39.740032            6    capitol-hill
3        NaN        NaN         NaN        NaN            6     five-points
4  3161788.0  1672521.0 -104.925197  39.678463            3       goldsmith

   IS_CRIME  IS_TRAFFIC
0         1           0
1         1           0
2         1           0
3         1           0
4         0           1
```

[56]:
```
#droping NaN values from the datafram
df.dropna(inplace=True)
df.head()
```

[56]:
```
                OFFENSE_TYPE_ID OFFENSE_CATEGORY_ID FIRST_OCCURRENCE_DATE  \
0                 theft-bicycle             larceny   6/8/2017 1:15:00 PM
7   burglary-residence-no-force            burglary  6/29/2017 8:00:00 AM
9          criminal-trespassing    all-other-crimes   6/7/2016 8:00:00 PM
17       theft-of-motor-vehicle          auto-theft  5/10/2016 3:00:00 PM
19  burglary-residence-by-force            burglary  1/31/2016 7:50:00 PM

    LAST_OCCURRENCE_DATE         REPORTED_DATE  \
0    6/8/2017 5:15:00 PM  6/12/2017 8:44:00 AM
7    7/5/2017 6:30:00 PM   7/5/2017 9:54:00 PM
9   6/8/2016 10:20:00 AM   6/8/2016 3:52:00 PM
17  5/10/2016 3:15:00 PM  5/10/2016 3:50:00 PM
19   2/1/2016 1:30:00 AM   2/1/2016 3:08:00 AM

                   INCIDENT_ADDRESS      GEO_X      GEO_Y     GEO_LON  \
0                      1705 17TH ST  3140790.0  1699792.0 -104.999264
7                  5004 N STUART ST  3128600.0  1712081.0 -105.042398
9                 3291 N OSCEOLA ST  3129744.0  1703432.0 -105.038484
17  E HARVARD AVE / S MILWAUKEE ST  3154183.0  1669827.0 -104.952274
19                1488 N MADISON ST  3155797.0  1694829.0 -104.945998

      GEO_LAT  DISTRICT_ID  NEIGHBORHOOD_ID  IS_CRIME  IS_TRAFFIC
0   39.753669           6    union-station         1           0
7   39.787581           1            regis         1           0
9   39.763822           1    west-highland         1           0
17  39.671197           3  university-park         1           0
19  39.739806           2    congress-park         1           0
```

[57]:
```
#Shape of the datafram
df.shape
```

[57]:
```
(48557, 14)
```

[58]:
```
df.columns = df.columns.str.replace(' ', '')
```

[60]:
```
df=df.rename(columns={"OFFENSE_TYPE_ID": "Offence Types", "GEO_LON":␣
↪"Longitude",
                      "GEO_LAT": "Latitude", "NEIGHBORHOOD_ID": "Neighborhood"})
df.head()
```

[60]:

	Offence Types	OFFENSE_CATEGORY_ID	FIRST_OCCURRENCE_DATE	LAST_OCCURRENCE_DATE	REPORTED_DATE	INCIDENT_ADDRESS	GEO_X	GEO_Y	Latitude	Longitude	DISTRICT_ID	Neighborhood	IS_CRIME	IS_TRAFFIC
0	theft-bicycle	larceny	6/8/2017 1:15:00 PM	6/8/2017 5:15:00 PM	6/12/2017 8:44:00 AM	1705 17TH ST	3140790.0	1699792.0	-104.999264	39.753669	6	union-station	1	0
7	burglary-residence-no-force	burglary	6/29/2017 8:00:00 AM	7/5/2017 6:30:00 PM	7/5/2017 9:54:00 PM	5004 N STUART ST	3128600.0	1712081.0	-105.042398	39.787581	1	regis	1	0
9	criminal-trespassing	all-other-crimes	6/7/2016 8:00:00 PM	6/8/2016 10:20:00 AM	6/8/2016 3:52:00 PM	3291 N OSCEOLA ST	3129744.0	1703432.0	-105.038484	39.763822	1	west-highland	1	0
17	theft-of-motor-vehicle	auto-theft	5/10/2016 3:00:00 PM	5/10/2016 3:15:00 PM	5/10/2016 3:50:00 PM	E HARVARD AVE / S MILWAUKEE ST	3154183.0	1669827.0	-104.952274	39.671197	3	university-park	1	0
19	burglary-residence-by-force	burglary	1/31/2016 7:50:00 PM	2/1/2016 1:30:00 AM	2/1/2016 3:08:00 AM	1488 N MADISON ST	3155797.0	1694829.0	-104.945998	39.739806	2	congress-park	1	0

[63]:
```
df['Neighborhood']=df['Neighborhood'].replace({'\-':' '}, regex = True)
df['Neighborhood']=df['Neighborhood'].str.title()
df.head()
```

[63]:

	Offence Types	OFFENSE_CATEGORY_ID	FIRST_OCCURRENCE_DATE	LAST_OCCURRENCE_DATE	REPORTED_DATE
0	theft-bicycle	larceny	6/8/2017 1:15:00 PM	6/8/2017 5:15:00 PM	6/12/2017 8:44:00 AM
7	burglary-residence-no-force	burglary	6/29/2017 8:00:00 AM	7/5/2017 6:30:00 PM	7/5/2017 9:54:00 PM
9	criminal-trespassing	all-other-crimes	6/7/2016 8:00:00 PM	6/8/2016 10:20:00 AM	6/8/2016 3:52:00 PM
17	theft-of-motor-vehicle	auto-theft	5/10/2016 3:00:00 PM	5/10/2016 3:15:00 PM	5/10/2016 3:50:00 PM
19	burglary-residence-by-force	burglary	1/31/2016 7:50:00 PM	2/1/2016 1:30:00 AM	2/1/2016 3:08:00 AM

```
0    6/8/2017 5:15:00 PM  6/12/2017 8:44:00 AM
7    7/5/2017 6:30:00 PM   7/5/2017 9:54:00 PM
9   6/8/2016 10:20:00 AM   6/8/2016 3:52:00 PM
17  5/10/2016 3:15:00 PM  5/10/2016 3:50:00 PM
19   2/1/2016 1:30:00 AM   2/1/2016 3:08:00 AM

                  INCIDENT_ADDRESS      GEO_X      GEO_Y    Latitude  \
0                      1705 17TH ST  3140790.0  1699792.0 -104.999264
7                  5004 N STUART ST  3128600.0  1712081.0 -105.042398
9                 3291 N OSCEOLA ST  3129744.0  1703432.0 -105.038484
17  E HARVARD AVE / S MILWAUKEE ST  3154183.0  1669827.0 -104.952274
19                1488 N MADISON ST  3155797.0  1694829.0 -104.945998

    Longitude  DISTRICT_ID     Neighborhood  IS_CRIME  IS_TRAFFIC
0   39.753669            6    Union Station         1           0
7   39.787581            1            Regis         1           0
9   39.763822            1    West Highland         1           0
17  39.671197            3  University Park         1           0
19  39.739806            2    Congress Park         1           0
```

[65]:
```
# create map for Denver City
map_denver = folium.Map(location=[39.744137, -104.950050], zoom_start=10)

# add markers to map
for lat, lng, neighborhood in zip(df['Longitude'], df['Latitude'],␣
 ↪df['Neighborhood']):
    label = '{}'.format(neighborhood)
    label = folium.Popup(label, parse_html=True)
    folium.CircleMarker(
        [lat, lng],
        radius=5,
        popup=label,
        color='blue',
        fill=True,
        fill_color='#3186cc',
        fill_opacity=0.7,
        parse_html=False).add_to(map_denver)

map_denver
```

[65]:
```
<folium.folium.Map at 0x1b3c49e51c8>
```

[ ]: