New York Crime Dashboard

Team Panini Head

Data Science Capstone Project

Exploratory Data Analytics Report

Date:

05/22/2021

Team Members:

Name: Ambrose Karella

Name: Janam Patel

Name: Naimish Bizzu

[The purpose of this report is to describe the exploratory data analytics. It includes five major sections:

1. Analyzing the basic metrics of variables: data types, size, descriptive statistics
2. Non-graphical and graphical univariate analysis: identifying unique value and counts, histogram, box plots, etc.
3. Missing value analysis and outlier analysis
4. Feature engineering and analysis: correlation analysis, dimensionality reduction, deriving new variables
5. Appendix]

Analysis the basic metrics of variables

[In this section, we identify all the variables in the dataset and conduct the basic metrics of the variables. What are the data types (numerical/categorical, discrete or continuous, ordinal or nominal) and size? Provide the descriptive statistics of the variables such as mean, standard deviation, min, max, percentiles, etc.]

There are no blanks or Null values in the final dataset that was analyzed.

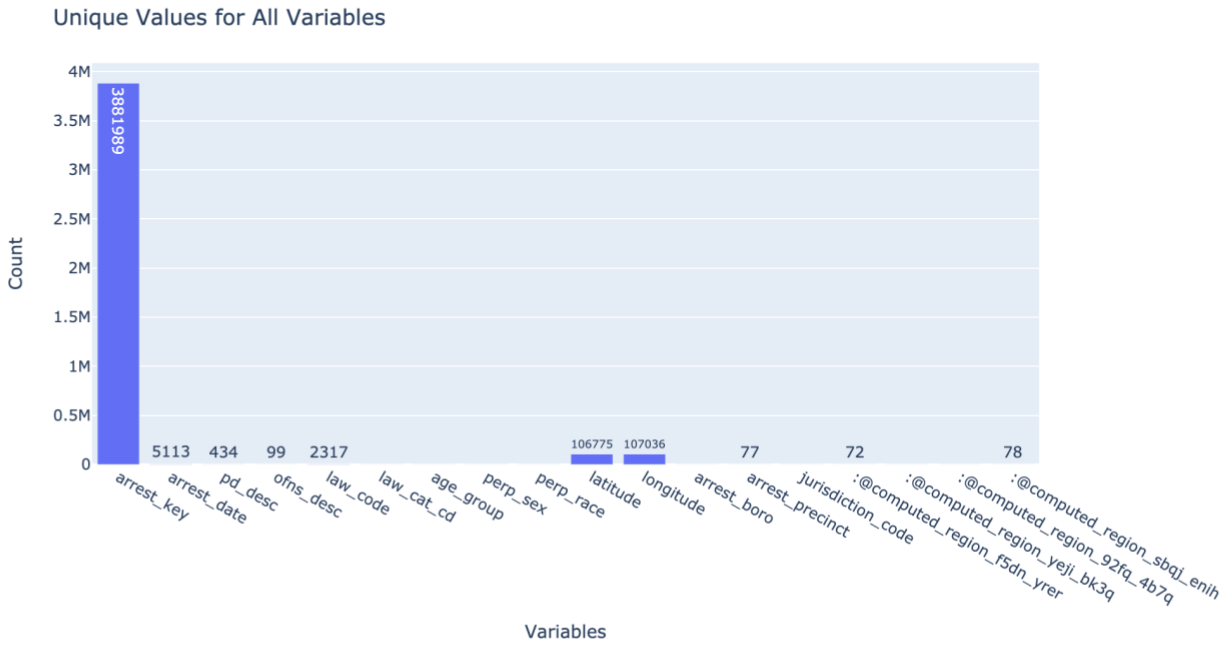
This is the Data we finally analyzed is as follows: -

- arrest_key – 3881989 unique values, numerical value where each value represents an unique crime case registered by the NYPD. Can't really provide any statistics on this column as the values are just random and are just there to identify cases.
- arrest_date – 5113 unique values, Dates ranges from 2006-01-01 to 2019-12-31. Finding statics for date is little troublesome, but we can use pandas quantile function which can do the job for us.
  - Min = 2006-01-01, the oldest record
  - Max = 2019-12-31, the newest record
  - 10th percentile = 2007-04-20
  - 25th percentile = 2009-02-22
  - *50th percentile or mean* = 2012-02-21
  - 75th percentile = 2015-06-07
  - 90th percentile = 2017-10-16
- pd_desc – 434 unique values, it's a categorical data but also can be said it's ordinal data but can't provide any real statistics since the values are in string format. Crimes in this column has levels and are specific,
  - For example - MARIJUANA, SALE 4 & 5
- ofns_desc – 99 unique values, it's a categorical data but can't provide any real statistics since the values are in string format. Crimes in this column are general term and aren't fully specific,
  - For example - DANGEROUS DRUGS
- law_code – 2317 unique values, it's a categorical data but can't provide any real statistics since the values contain both alphabets and numbers as well. They're Law code charges corresponding to the NYS Penal Law, VTL and other various local laws

- law_cat_cd – 5 unique values, it’s a categorical data but can’t provide any real statistics since they are string value. The values represent level of offense like ‘f’ for Felony.
- age_group – 5 unique values, it’s a categorical data and can’t provide any real statistics since they are string value. The values represent perpetrator’s age in category.
- perp_sex – 2 unique values, it’s a categorical data can’t provide any real statistics since they are string value and just represent M (male) or F (female).
- perp_race – 8 unique values, it’s a categorical data can’t provide any real statistics since they are string value and just represents perpetrator’s race description
- latitude – 106775 unique values, Location data
- longitude – 107036 unique values, Location data
- arrest_boro – 5 unique values, it’s a categorical data of where perpetrator was arrested and can’t provide any statistics since they are string value.
- arrest_precinct – 77 unique values, it’s a numerical data where values represent Precinct where the arrest occurred, so can’t provide any statistics..
- jurisdiction_code – 27 unique values, it’s a numerical data where values represent Jurisdiction responsible for arrest, so can’t provide any statistics.
- :@computed_region_f5dn_yrer – 72 unique values, Community District
- :@computed_region_yeji_bk3q – 6 unique values, Borough Boundaries
- :@computed_region_92fq_4b7q – 52 unique values, City Council Districts
- :@computed_region_sbqj_enih – 78 unique values, Police Precincts

Non-graphical and graphical univariate analysis

[In this section, we identify the list and number of unique values for each variables and provide the histogram and box plots to understand the distribution of the data.]



As you can see, there are some variables that don't show any values. The reason being that their value for unique elements in that variable is very low, like perp_sex has 2 unique values, and arrest_boro has 5 unique values.

Missing value analysis and outlier analysis

[In this section, we identify the missing values and outliers and determine how we handle these values before analysis.]

- We had removed the majority of missing and Null values from our data during our preprocessing since there weren't that many and we thought it'd be easier to analyze data where we don't have missing or Null values.

Feature engineering and analysis

[In this section, we identify the variables that are useful for predictive modeling and machine learning through correlation analysis. You may also reduce the dimension or derive new variables so that the predictive modeling can be more efficient and effective.]

Some variables that can be useful for predictive modeling and machine learning through correlation analysis are: -

- arrest_precinct
- Latitude
- Longitude
- :@computed_region_f5dn_yrer – Community District
- :@computed_region_yeji_bk3q – Borough Boundaries

- :@computed_region_92fq_4b7q – City Council Districts
- :@computed_region_sbqj_enih – Police Precincts

They can be important, to determine if a crime occurs in specific location, then which Community, City Council Districts and Borough would it fall under and which Police Precinct is most likely to answer to the crime. We can also, specify some type of crime and see which locations might have kind of crime occur and during which time of the year season wise or month wise.

Appendix

[Provide the code or pseudo code, and any other information in the appendix here.]

- Data Collector using concurrent futures.

```
with concurrent.futures.ProcessPoolExecutor() as executor:

        results = [executor.submit(get_data, ij) for ij in range(5012)]

        for f in concurrent.futures.as_completed(results):
            fdb = pd.concat([fdb, f.result()])
```

- Data Reading and downloading from SQL

```
# pulls clean data from repo and saves locally
if path.exists("crime.dat"):
    print("Found crime data. Loading as pandas df")
    fh = open("crime.dat", 'rb')
    df = pickle.load(fh)
    fh.close()
else:
    print("Crime data not downloaded, saving file in root dir as 
crime.dat")
    print("Downloading 1.2GB, this may take a while...")
    df = pd.read_sql('SELECT * FROM crimeTable', con=engine)
    fh = open('crime.dat', 'wb+')
    pickle.dump(df, fh)
    fh.close()
```

- Animation of a Plotly Graph showing Crime in each Borough throughout the years.

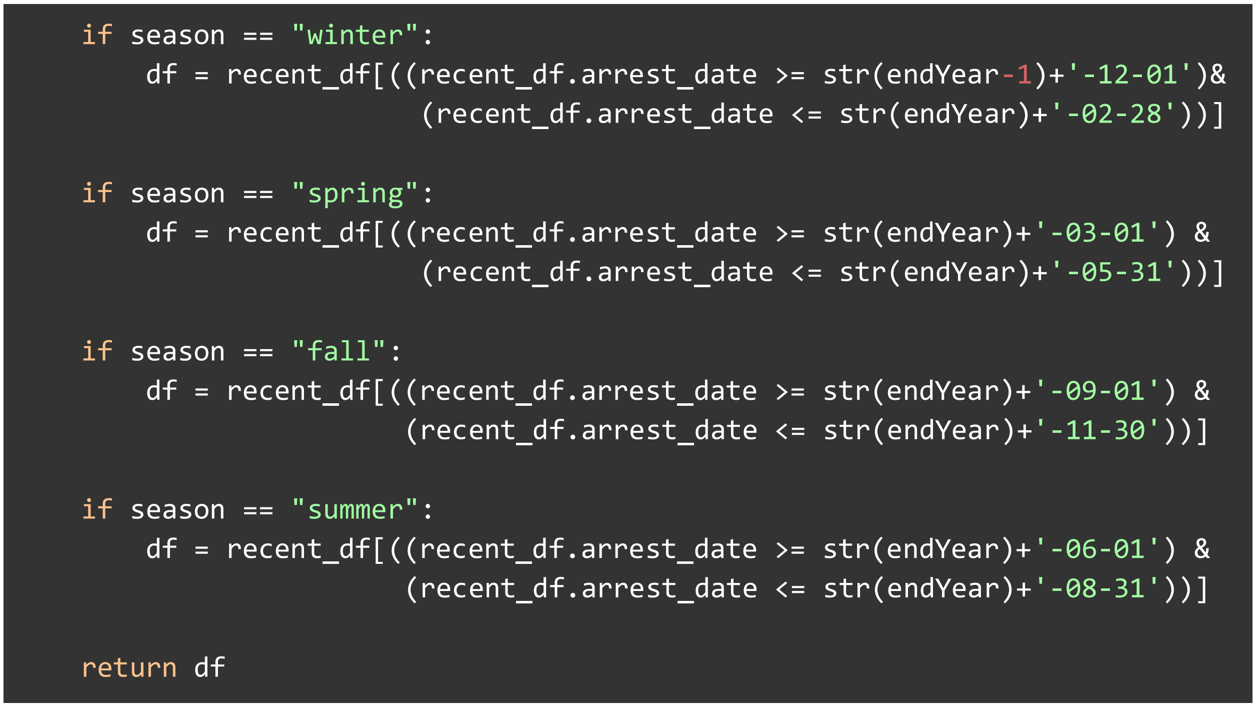
```
boro_co =
list(df1.groupby(['arrest_year','arrest_boro']).count()['arrest_key'])
boro = list(sorted(df1.arrest_boro.unique()))*14
years = list(np.repeat(list(sorted(df1.arrest_year.unique())),5))

fig = px.scatter(x = years, y = boro_co,
                 animation_frame = years,
                 animation_group = boro,
                 color = boro,
               size = boro_co,
               range_x = [min(years)-1,max(years)+1],
               range_y = [6000,95000]
                )

fig.update_layout(
    title_text='Borough Crime throughout the years',
    xaxis_title="Years",
    yaxis_title="Cases")
fig.update_xaxes(tick0=1, dtick=1)

fig.show()
```

- CDF Graph

```
boro_group = df1[df1['arrest_boro']==i]

# get the crime count per day.
sd = list(boro_group.groupby('arrest_date').count()['arrest_key'])

# sort the counts for cdf
monx = np.sort(sd)

# define the probability values
mony = np.arange(1,len(monx)+1)/len(monx)
```

- Function to determine and create Seasons dataframe

```
recent_df = df.loc[(df['arrest_date'] >= '2014-12-01')]

def getSeasonData(season,endYear):
```



```
    if season == "winter":
        df = recent_df[((recent_df.arrest_date >= str(endYear-1)+'-12-01')&
                         (recent_df.arrest_date <= str(endYear)+'-02-28'))]

    if season == "spring":
        df = recent_df[((recent_df.arrest_date >= str(endYear)+'-03-01') &
                         (recent_df.arrest_date <= str(endYear)+'-05-31'))]

    if season == "fall":
        df = recent_df[((recent_df.arrest_date >= str(endYear)+'-09-01') &
                        (recent_df.arrest_date <= str(endYear)+'-11-30'))]

    if season == "summer":
        df = recent_df[((recent_df.arrest_date >= str(endYear)+'-06-01') &
                        (recent_df.arrest_date <= str(endYear)+'-08-31'))]

    return df
```

Table of Contributions

The table below identifies contributors to various sections of this document.

	Section	Writing	Editing
1	**Analysis the basic metrics of variables**	**Naimish, Janam**	**Ambrose**
2	**Non-graphical and graphical univariate analysis**	**Naimish, Janam**	**Ambrose**
3	**Feature engineering and analysis**	**Naimish, Janam**	**Ambrose**
4	**Appendix**	**Naimish, Janam**	**Ambrose**

Grading

The grade is given on the basis of quality, clarity, presentation, completeness, and writing of each section in the report. This is the grade of the group. Individual grades will be assigned at the end of the term when peer reviews are collected.