**Project name: Pittsburgh Police Arrest Data**

**Date: 07-Apr-2019**

**Author: Niloofar Zarifi**

Arrest data contains information on people taken into custody by City of Pittsburgh police officers. More serious crimes such as felony offenses are more likely to result in an arrest. All data is public with the exception of sex crimes, which are reported at the police zone level. This dataset does not contain incidents that solely involve other police departments operating within the city. There is a clear guideline (referring to page. 2) that eases the understanding of the information reported for each column. The Pittsburgh Police Arrest Data is up to date and it was downloaded from https://catalog.data.gov/dataset/pittsburgh-police-arrest-data on March 26, 2019. Therefore I kept the data from January 2016 as of March 2019. I briefly explain the analysis procedures to the following objectives:

1. ARRESTTIME

We are able to answer the following questions: Which year, month, season, hour, weekdays have the most reports? Or if the crime season has constant behavior?

In order to study the ‘arrest time’ first, I defined the function to split the date to year, month, day, hours... etc. Then I mostly used value\_counts() function and matplotlib to visualize the data. To explore the correlation of year and month of the crime I used matplotlib package instead of year and month columns. The reason is that the arrest date is not in ascending order for 2016 which causes the wrong date order in histogram plot. To avoid sorting the date I applied one of matplotlib packages to convert arrest date to index and count year and month in decent order. Please find more details in the arresttime.py file

1. RACE

To investigate the statistic of the RACE, I used value\_counts() and changed the race dictionary to list the percentage of the crime. The pie chart in the proper format was presented as well. The details of the code could be found in RACE.py

1. GENDER and AGE

In this section, after cleaning the missing/wrong data, I presented the rate of crime based on gender between 2016 to 2019 and if the crime reports increases for men and women. The correlation of age, year and gender was studied in whisker and box plot and age group was defined to categorize the crime rate for each individual age group. More details is shown in GENDER-AGE.py

1. COUNCIL DISTRICT and WORKD DIVISION

Here I show which council district and public works division are the busiest.

1. ARRESTLOCATION, OFFENSES and INCIDENTLOCATION

I investigate which neighbourhoods have the most crime records and if arrest and incident location are occurred in same location. I also show in detail the top 5 most common offenses that cause arrestment in Pittsburgh. The python code was stored in COUNCIL.py

Conclusion:

Data preprocessing is an important step in the data analysis. In a real world we deal with incomplete, noisy, missing/wrong values in data set. A careful preprocessing could prevent poor analysis. Visualizing data facilitate the conception of the data but creating the publishable format is quite challenging.

**The list of the attached documents:**

Python codes: arresttime.py, RACE.py, GENDER-AGE.py, COUNCIL.py

Pittsburgh.csv and all plots

Niloofar\_zarifi.pptx

Crime data guide:

|  |  |
| --- | --- |
| ARREST\_ID | Unique identifier for each arrest incident |
| CCR | Incident Number |
| AGE | Age of suspect |
| GENDER | Gender of suspect |
| RACE | Race of suspect |
| ARREST\_DATE | Date of Arrest |
| ARREST\_TIME | Time of Arrest |
| ARRESTLOCATION | Location of Arrest |
| OFFENSES | All offenses listed for the incident |
| INCIDENTLOCATION | Incident Location |
| NEIGHBORHOOD | Neighborhood where incident occured |
| ZONE | Police Zone where Incident occurred |
| X | The geocoded X coordinate of the incident location (using county geocoder) |
| Y | The geocoded Y coordinate of the incident location (using county geocoder) |