**Purpose**

Is there a correlation between gun violence, the cities where it occurs and the demographics within those locations? Also, does the current political leader of those cities influence these potential correlations?

**Methodology:**

**Data Sources**

Extract a dataset from Kaggle documenting Gun Violence Mass Shootings from the past 50 years and used the pd.read.csv function.

Extract a dataset from US Government resources documenting congressional member history used the response module and for loop to read the json object.

Extract a dataset from a geocoding API resource which retrieved the corresponding congressional data related to global coordinates and used the pd.read.csv function.

**Data Cleaning**

Clean the csv files by removing duplicate entries, dropping unneeded columns, dropping any Nan records in data.

Cleaning included ensuring all units across columns were uniform, additional columns were added as necessary including the split of multiple records, namely dates and city/states. Also, additional columns were concatenated where appropriate in order to be used in the creation of the future database.

Equally important, data was evaluated and outliers were questioned. We ensured the data was unbiased, noise was removed and erroneous values were excluded.

We used Pandas to convert our cleaned data from data frames to csv files so that we may prepare to import data into our database.

**Creating a Database**

Using PostgreSQL, we create a gun violence/mass shooting database. Our tables were: Legislator\_Data, Congressional\_District\_Data, and Mass\_Shooting

We experienced some challenges importing data into the tables. These challenges were due to the data types listed on the columns for the tables as they didn’t match the data types of the csv files. Once identified we were able to rectify the cause of error and successfully import the data. We also learned best practices about how to name columns in data frames and tables.

Originally, we thought our primary keys would be our global coordinates. Once in pgAdmin, we found there were duplicates across the three tables. We also then found duplicates across the Incident name, and incident date disenabling us from creating a primary key. Our foreign keys were the congressional districts, connecting data and allowing access across all three tables. We were still able to join our tables without primary keys on the aforementioned columns.

**Sharing our Database for Use**

Flask

HTML