DSC170 Final Project

March 20, 2024

1 1. Impact of Socioeconomic and Environmental Factors on Shootings in California

1.1 2. Team Members: Rihana Mohamed (A16668393), Ricky Miura (A1628355), Saachi Shenoy (A16296514)

1.2 3. Questions(s) addressed and target audience

Living in the United States of America, the term mass shooting and shooting comes up in news very often. Since it is such an issue facing society we wanted to analyze whether factors affected it and specifically use geospatial analysis to figure out the factors that affect shootings. Our intended audience was the general public of the United States just to understand whether there are factors that affect a shooting based on previous data. Since the data is public we thought analyzing shootings would be informative and would ease people's minds, since shootings tend to happen at a high rate. Our general question was: Does location (distance from police stations) and socioeconomic characteristics (demographics, income, mental health, incarceration) impact the whether a shooting is a mass shooting and how many victims would be hurt?

Our project overall has not changed since we analyzed shootings. The main thing that changed was our analysis shifted from rates of shootings happening to mass shootings happening and the death toll. The reason this changed was because our data changed. We initially thought we could find the likelihood of shootings happening but there was no data to support this. So we decided to instead use the data we had on number killed and number injured to figure out victim toll as well as classifying a shooting as a mass shooting.

1.3 4. Background and Literature

- Mass Shootings: Why is it an issue Between the year 2015 and 2022 19,000 people were victims (killed or wounded) because of mass shootings. Mass shootings, according to the FBI, are when three or more people are killed. This article talks about how several countries in the world have similar rates of depression and mental illness. The issue with our country is the easy access to guns and how there aren't as many background checks when selling guns. This article is relevant in our study because we can see what the target audience would be. Since it is such a prevelant issue in the United States it makes sense to look at shootings there because we acknowledge that its such an issue.
- Income inequality and mass shootings in the United States Since what we are trying to look for is the presence of socioeconomic differences between different areas and whether that made a difference in a shooting occuring. This article talks about the income differences and whether that affected the likelihood of mass shootings. Before starting our analysis on shootings we

wanted to make sure that income did affect it and wanted to find prior research on this. This paper was able to find a 43% more likelihood of a shooting happening in a lower income area.

- Mass Shootings Rise in California amid National Surge While looking through our data we wanted to decide which areas to specifically look at. Our dataset had shootings for the entire country and within our dataset we did see a huge amount in California. However, this article further talks about why shootings in California happen so much. This article talks about how a mass shooting happens in California every 4 days in 2023. This is alarming and is another reason we wanted to do our analysis on mass shootings. Shootings are such a huge problem in America right now and people are worried about it.
- Is There a Link Between Mental Health and Mass Shootings? When looking at articles regarding mass shootings, we always noticed that the shooters tend to have specific mental illnesses. We wanted to include this in our research specifically based on location. We thought that places with higher mental health issues such as depression would have higher shootings occuring. This article talks about how mental illness does have a link to whether an individual becomes a shooter. According to this article "a much larger number of mass shootings (about 25%) are associated with non-psychotic psychiatric or neurological illnesses, including depression".

1.4 5. Imported Packages

Python Libraries: - pandas: python library for data manipulation and analysis, providing data structures like DataFrame for handling structured data - numpy: package for numerical computing in Python, also supports large, multi-dimensional arrays and matrices, along with a collection of mathematical functions to operate on these arrays - warnings: a module that manages warnings, allowing developers to control how warnings are displayed and handled during code execution - re: a module used for working with regular expressions, enabling pattern matching and text manipulation tasks. - matplotlib.pyplot: a plotting library that provides a MATLAB-like interface for creating static, interactive, and animated visualizations, making it suitable for data exploration and presentation - shapely.geometry: a library for geometric operations, allowing manipulation and analysis of geometric objects like points, lines, and polygons

Arcgis Libraries: - arcgis.gis: a module providing access to the ArcGIS Online and ArcGIS Enterprise GIS platforms, allowing users to interact with GIS resources, perform spatial analysis, and manage content. - geopandas: a pandas like module that adds support for working with geospatial data, integrating seamlessly with other Python GIS libraries like Fiona and Shapely. - arcgis.raster: a module within the ArcGIS API for python specifically designed for working with raster data, enabling raster manipulation, analysis, and visualization. - arcgis.geometry: a module providing tools for working with geometric objects and performing spatial operations such as buffering, clipping, and intersection within the ArcGIS ecosystem. - arcgis.geocoding: a module offering geocoding capabilities, allowing users to convert addresses or place names into geographic coordinates (latitude and longitude) and vice versa, essential for mapping and spatial analysis tasks.

This is similar to the list that we brainstormed for our project proposal however we added arcgis geocoding to this list

[42]:

```
#### import arcgis
from arcgis.gis import GIS
import pandas as pd
import numpy as np
import warnings
from shapely.geometry import Point
import geopandas as gpd
import re
import matplotlib.pyplot as plt
from arcgis.raster import *
from arcgis.geometry import *
from arcgis.geocoding import *
from sklearn.model_selection import train_test_split
from sklearn.linear_model import LinearRegression, LogisticRegression
from sklearn.metrics import mean_squared_error
```

1.5 6. Data Sources

- Shooting Dataset: The gun violence dataset from Kaggle provides detailed information on incidents including the number of casualties, location by city, and additional contextual data. It offers valuable insights into trends and patterns of gun-related incidents for analysis and research purposes.
- Counties Feature Layer: source id: '8713ced9b78a4abb97dc130a691a8695': This data represents a feature layer containing the border information for counties within California, facilitating spatial analysis and visualization of gun violence incidents within specific county boundaries
- Police Station Feature Layer:source id: '9e1d7326d36c4725b28fb49fb638b8f5': This data corresponds to a feature layer pinpointing the locations of police stations, which can be utilized to explore the proximity of law enforcement resources to areas affected by gun violence incidents
- Depression Data: The database from the Mental Health America (MHA) website offers comprehensive state and county-level data on depression rates, measured per 100,000 people, providing valuable insights into the prevalence and distribution of depression across the United States. Additionally, it features an interactive map interface, enabling users to visualize and explore the regional variations in depression rates, aiding in targeted interventions and resource allocation efforts.
- Income, population and Incarceration Data The dataset available through the National Institutes of Health's HD Pulse platform offers comprehensive data on income, population demographics, and incarceration rates at the state level, providing insights into socio-economic disparities and criminal justice involvement. Understanding the intersection of these factors can aid in identifying areas with higher vulnerability to various social challenges, including gun violence, and inform targeted interventions and policy initiatives aimed at addressing underlying systemic issues.
- Unemployment Data The unemployment dataset obtained from a website provides valuable information on the unemployment rates across different counties, offering insights into regional economic conditions and disparities.This data can also serve as a crucial factor in understanding the socio-economic factors associated with gun violence, as areas with higher unemployment rates may experience increased levels of social tension, economic stress, and crime, potentially contributing to higher rates of gun-related incidents. Understanding the

relationship between unemployment and shootings can inform targeted intervention strategies aimed at addressing underlying socio-economic disparities and mitigating the risk of gun violence in vulnerable communities.

For our proposal we only had our shooting data identified but we realized while we were working on the project that we needed more data to identify socioeconomic trends. We started off trying to find it per city since the dataset is per city but realized that data isn't as accessible so we found it for counties and geoencoded the cities to counties

1.6 7. Cleaning the Data

We had to individually clean up each data source that we used for this analysis. They are documented in markdown cells below. The overall datasets was cleaning of the dataframe themselves and fixing wording and the way the data was formatted. The cleaning process was a lot more than what we anticipated when we were brainstorming for our proposal. This is because with our proposal we wanted to just use the shooting dataset since it had longitude and latitude, so we thought it would be minimal cleaning.

[2]:

```
gis = GIS(username="dsc170wi24_23")
```

```
Enter password: ········
```

This is the arcgis feature layer we used for counties. The point of this layer was to geocode the cities to counties. We used the join feature to spatially intersect the counties and the latitiude and latitude given per incident. With this cleaning we were able to figure out which county each shooting occurred in.

[3]:

```
#counties data
counties_layer = gis.content.get('8713ced9b78a4abb97dc130a691a8695')
```

In order to just look at California we had to filter and make sure the state we were looking at for the shooting incidents were only in California.

[4]:

```
#shooting data
df = pd.read_csv('gun-violence-data_01-2013_03-2018.csv')
df = df[['incident_id', 'date', 'state', 'city_or_county', 'n_killed',
       'n_injured' ,'latitude', 'longitude']]
df = df.dropna(subset = ['latitude', 'longitude'])
df['year'] = pd.to_datetime(df['date']).dt.year
df['month'] = df['date'].apply(lambda x: x[5:7])
#just lookinga at california
california = df[df['state'] == 'California']
```

[5]:

```
df.head(3)
```

[5]:

```
   incident_id        date         state city_or_county  n_killed  n_injured  \
0       461105  2013-01-01  Pennsylvania     Mckeesport         0          4
1       460726  2013-01-01    California      Hawthorne         1          3
2       478855  2013-01-01          Ohio         Lorain         1          3
```

```
   latitude  longitude  year month
0   40.3467   -79.8559  2013    01
1   33.9090  -118.3330  2013    01
2   41.4455   -82.1377  2013    01
```

[6]:
```
counties_df = pd.DataFrame.spatial.from_layer(counties_layer.layers[0])
```

[7]:
```
sdf_california_id = pd.DataFrame.spatial.
 ↪from_xy(california[['longitude','latitude', 'incident_id']] , x_column =␣
 ↪'longitude', y_column='latitude', sr = 4326)
```

[8]:
```
shootings = sdf_california_id.spatial.to_featurelayer(title='shooting',␣
 ↪gis=gis, tags="finalproj")
```

[9]:
```
#joining counties and cities to find which counties the cities fall into
joined = arcgis.features.analysis.join_features(target_layer=shootings,
                                                join_layer=counties_layer,
                                               ␣
 ↪spatial_relationship="Intersects",
                                               ␣
 ↪output_name="joined_county_point_layer_9")
```

```
{"cost": 16.061}
```

[10]:
```
join_item = gis.content.get(joined['id'])
join_result_layer = join_item.layers[0]
```

[11]:
```
feat = join_result_layer.query(where="1=1", out_fields="*",␣
 ↪return_geometry=False)
```

[12]:
```
feature_attributes = [feature.attributes for feature in feat.features]

# Convert to DataFrame
shooting_by_county = pd.DataFrame(feature_attributes)
```

We had to make sure the espg was the same for the join we were doing.

[13]:
```
counties_df = pd.DataFrame.spatial.from_layer(counties_layer.layers[0])
counties_gpd = gpd.GeoDataFrame(counties_df, geometry='SHAPE')
counties_gpd.set_crs("epsg:3857", inplace=True)
counties_gpd = counties_gpd.to_crs("epsg:4326")
```

[14]:
```
gpd_county = counties_gpd[['COUNTY_NAME', 'SHAPE']].drop_duplicates(subset =␣
 ↪['COUNTY_NAME'])
```

Prior to importing the dataset for income we had to go through each county and clean the data. There were no null values but all the counties were worded differently than the wording for the rest of our data. So in order to clean this up we had to manually go into the data and reword the wording for all the counties.

[15]:

```
#income per county
income = pd.read_csv('county_income_population.csv')
```

The depression data was an interactive map, so we had to filter by county within the state we were interested in (California). We were then able to export an excel file. Again with this data the dataset county names were not clean and we had to manually go in and rename each county.

[16]:

```
#depression count per 100k per county
depression = pd.read_csv('depression_per_county.csv')
depression = depression[['County Name', 'Calc Total Depression Responses per␣
 ↪100K']]
```

The incarceration rate was found at the same source as the income. They both were cleaned the same way.

[17]:

```
#incarceration per county
incarceration = pd.read_csv('incarceration.csv')
```

We weren't able to find a dataframe form of the unemployment rate per county. We were however to find the data for each county so we manually decided to create a dataframe from the website described above. From this website we were able to find the unemployment rates and build a dataframe from creating an array of county names and unemployment rates and merging them.

[18]:

```
#unemployment rate per county
county_names = [
    "ALAMEDA", "ALPINE", "AMADOR", "BUTTE", "CALAVERAS", "COLUSA", "CONTRA␣
 ↪COSTA",
    "DEL NORTE", "EL DORADO", "FRESNO", "GLENN", "HUMBOLDT", "IMPERIAL",␣
 ↪"INYO", "KERN", "KINGS",
    "LAKE", "LASSEN", "LOS ANGELES", "MADERA", "MARIN", "MARIPOSA",␣
 ↪"MENDOCINO", "MERCED", "MODOC",
    "MONO", "MONTEREY", "NAPA", "NEVADA", "ORANGE", "PLACER", "PLUMAS",␣
 ↪"RIVERSIDE", "SACRAMENTO",
    "SAN BENITO", "SAN BERNARDINO", "SAN DIEGO", "SAN FRANCISCO", "SAN␣
 ↪JOAQUIN", "SAN LUIS OBISPO",
    "SAN MATEO", "SANTA BARBARA", "SANTA CLARA", "SANTA CRUZ", "SHASTA",␣
 ↪"SIERRA", "SISKIYOU",
    "SOLANO", "SONOMA", "STANISLAUS", "SUTTER", "TEHAMA", "TRINITY", "TULARE",␣
 ↪"TUOLUMNE", "VENTURA",
    "YOLO", "YUBA"
]

unemployment_rates = [
    "5.7%", "5.0%", "6.6%", "6.2%", "6.6%", "5.4%", "19.3%", "5.0%", "7.2%", "5.
 ↪0%", "8.8%", "8.0%",
    "5.9%", "17.8%", "4.8%", "9.5%", "9.9%", "7.4%", "7.1%", "5.9%", "8.7%", "4.
 ↪1%", "7.0%", "6.1%",
    "10.9%", "10.2%", "4.3%", "10.5%", "4.6%", "4.8%", "4.2%", "4.5%", "11.0%",␣
 ↪"5.5%", "5.3%",
    "7.5%", "5.4%", "4.7%", "4.0%", "7.5%", "4.2%", "3.7%", "5.6%", "4.3%", "7.
 ↪4%", "6.4%", "6.2%",
    "8.7%", "5.7%", "4.5%", "7.6%", "9.8%", "7.0%", "7.5%", "11.8%", "6.2%", "5.
 ↪2%", "6.1%", "8.3%"
]

county_names = [name.title() for name in county_names]
unemployment_rates = [float(rate.strip('%')) for rate in unemployment_rates]
county_unemployment_dict = dict(zip(county_names, unemployment_rates))
unemployment = pd.DataFrame.from_dict(county_unemployment_dict, orient='index').
 ↪reset_index().rename(columns = {'index': 'County', 0:'Unemployment Rate'})
```

We then merged all the different dataframes we constructed by county names to get a full dataset with our socioeconomic information as well as our shooting information.

[19]:

```
#merge all the datasets
df_1 = shooting_by_county.merge(california, left_on = 'incident_i', right_on =␣
 ↪'incident_id').drop_duplicates(subset = ['incident_id'])
df_2 = df_1.merge(unemployment, left_on = 'COUNTY_NAME', right_on = 'County',␣
 ↪how ='inner')
df_3 = df_2.merge(income, left_on = 'COUNTY_NAME', right_on = 'county', how =␣
 ↪'inner')
df_4 = df_3.merge(incarceration, left_on = 'COUNTY_NAME', right_on = 'county',␣
 ↪how = 'inner')
df_5 = df_4.merge(depression, left_on = 'COUNTY_NAME', right_on = 'County Name')
```

Our final data is the proximity of a police station to the point that the shooting happened. We wanted to see if there was a relationship between this since a lot of shootings are classified as suicide shootings, where the shooter hopes or plans to get caught. We saw if the police station was within 1 mile of the shooting and classified it within the buffer that we defined.

[21]:

```
#police stations
police = gis.content.get('9e1d7326d36c4725b28fb49fb638b8f5').layers[0]
police_wrong_crs = pd.DataFrame.spatial.from_layer(police)
correct_crs = pd.DataFrame.spatial.from_xy(police_wrong_crs[['X', 'Y']] ,␣
 ↪x_column = 'X', y_column='Y', sr = 4326)
sdf_police_fl = correct_crs.spatial.to_featurelayer(title ='USA Police␣
 ↪Station', gis=gis, tags="Final-Project-EDA")
shooting_spat_df = pd.DataFrame.spatial.from_xy(df_5[['longitude_x',␣
 ↪'latitude_x', 'OBJECTID']] , x_column = 'longitude_x',␣
 ↪y_column='latitude_x', sr = 4326)
sdf_shooting_fl = shooting_spat_df.spatial.to_featurelayer(title ='Shootings␣
 ↪for Station', gis=gis, tags="Final-Project-EDA_1")
```

```
from arcgis.features import analysis
shootings_joined = analysis.join_features(target_layer = sdf_shooting_fl,␣
 ↪join_layer = sdf_police_fl,join_operation = "JoinOneToOne",
                               join_type = "LEFT", spatial_relationship =␣
 ↪"Intersects", spatial_relationship_distance = 5280 ,
                               spatial_relationship_distance_units = "Feet",␣
 ↪output_name = "Intersection_1_mile_buffer_80")
join_item = gis.content.get(shootings_joined['id'])
join_result_layer = join_item.layers[0]
feat = join_result_layer.query(where="1=1", out_fields="*",␣
 ↪return_geometry=False)
bool_police = [feature.attributes.get('Join_Count', 0) > 0 for feature in feat]
df_5['within_police'] = bool_police
df_5['within_police'] = df_5['within_police'].astype(int)
```

```
{"cost": 39.585}
```

We created a new variable called total victims that combined the number of injuries and number killed and created our final dataframe.

[22]:

```
df_6 = df_5.merge(gpd_county, left_on = 'COUNTY_NAME', right_on = 'COUNTY_NAME')

final_df = df_6[['within_police','longitude_x', 'latitude_x',␣
 ↪'COUNTY_NAME','incident_id', 'date', 'state', 'n_killed', 'n_injured',␣
 ↪'year', 'month',  'Unemployment Rate','income', 'population', 'incarceration␣
 ↪rate (per 100000)','Calc Total Depression Responses per 100K', 'SHAPE']].
 ↪rename(columns = {'longitude_x': 'longitude', 'latitude_x': 'latitude',␣
 ↪'COUNTY_NAME': 'county','Unemployment Rate': 'unemployment_rate',␣
 ↪'incarceration rate (per 100000)': 'incarceration_rate','Calc Total␣
 ↪Depression Responses per 100K': 'total_depression_responses', 'SHAPE':␣
 ↪'geometry'})
final_df['total_victims'] = final_df['n_killed'] + final_df['n_injured']
final_df['total_depression_responses'] = final_df['total_depression_responses'].
 ↪str.replace(',', '')
final_df['total_depression_responses'] = final_df['total_depression_responses'].
 ↪astype(float)
```

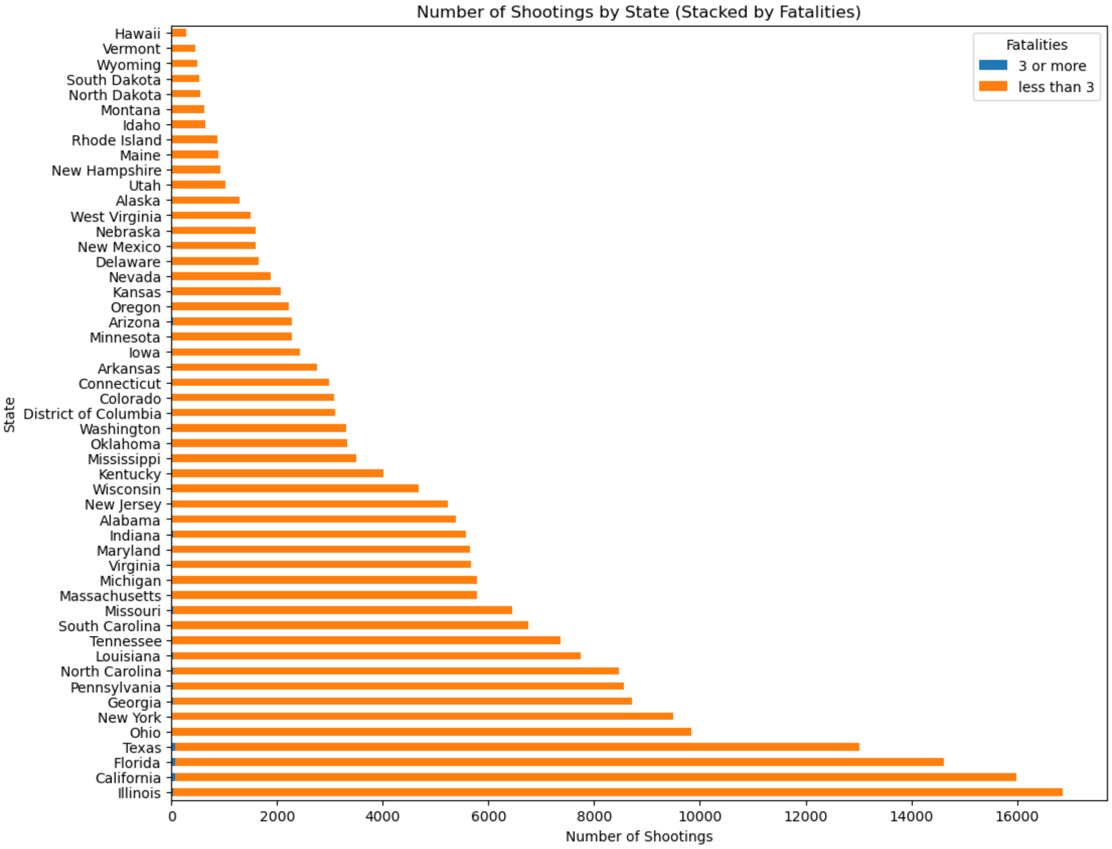
1.7 Exploratory Data Analysis (EDA)

1.7.1 Choosing our Area of Interest (AOI)

“…U.S. statute (the Investigative Assistance for Violent Crimes Act of 2012) defines a “mass killing” as ‘3 or more killings in a single incident.”’ (https://www.britannica.com/topic/mass-shooting)

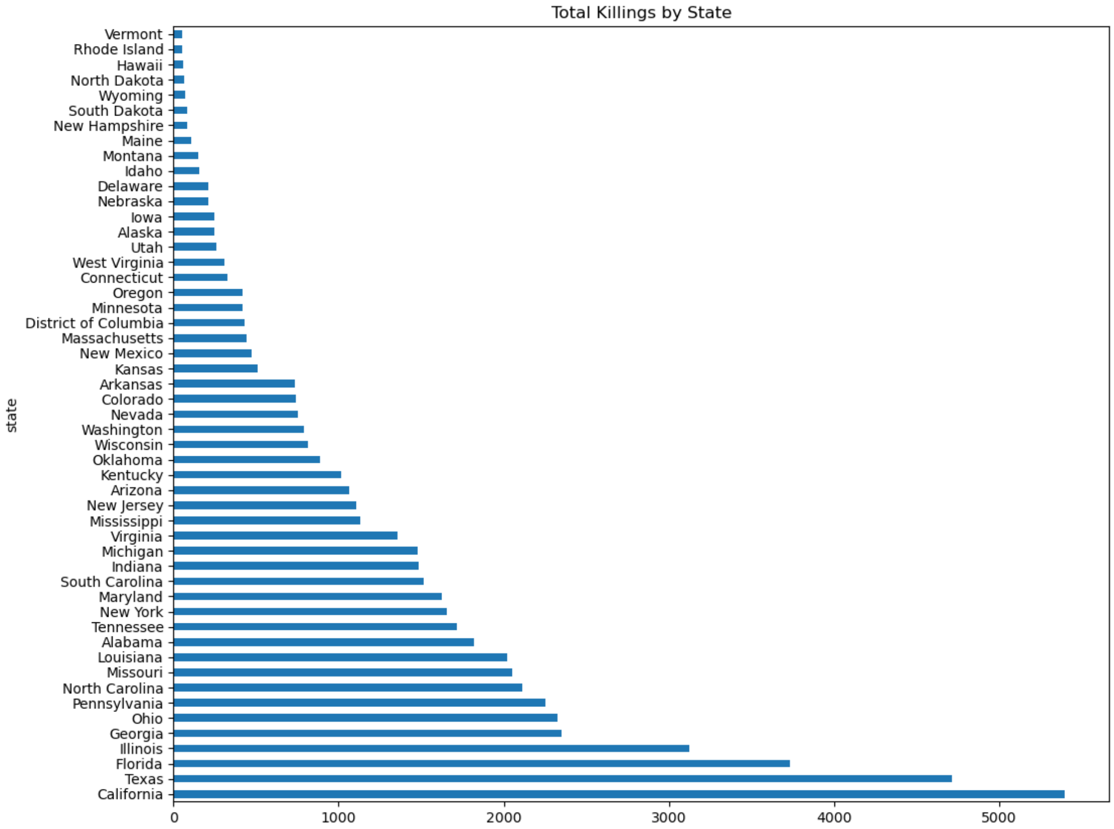
[23]:

```
df['fatal_shooting'] = df['n_killed'].apply(lambda x: '3 or more' if x >= 3␣
 ↪else 'less than 3')
shooting_counts = df.groupby(['state', 'fatal_shooting'])['incident_id'].
 ↪count().unstack()
shooting_counts = shooting_counts.fillna(0)
shooting_counts['total_shootings'] = shooting_counts.sum(axis=1)
shooting_counts = shooting_counts.sort_values(by='total_shootings',␣
 ↪ascending=False)
shooting_counts.drop(columns='total_shootings').plot(kind='barh', stacked=True,␣
 ↪figsize=(12, 10))
plt.xlabel('Number of Shootings')
plt.ylabel('State')
plt.title('Number of Shootings by State (Stacked by Fatalities)')
plt.legend(title='Fatalities')

plt.show()
```



```
[24]: df.groupby('state')['n_killed'].sum().sort_values(ascending=False).
 ↪plot(kind='barh', figsize=(12,10), title='Total Killings by State')
```

```
[24]: <Axes: title={'center': 'Total Killings by State'}, ylabel='state'>
```



1.7.2 Investigating California Shootings

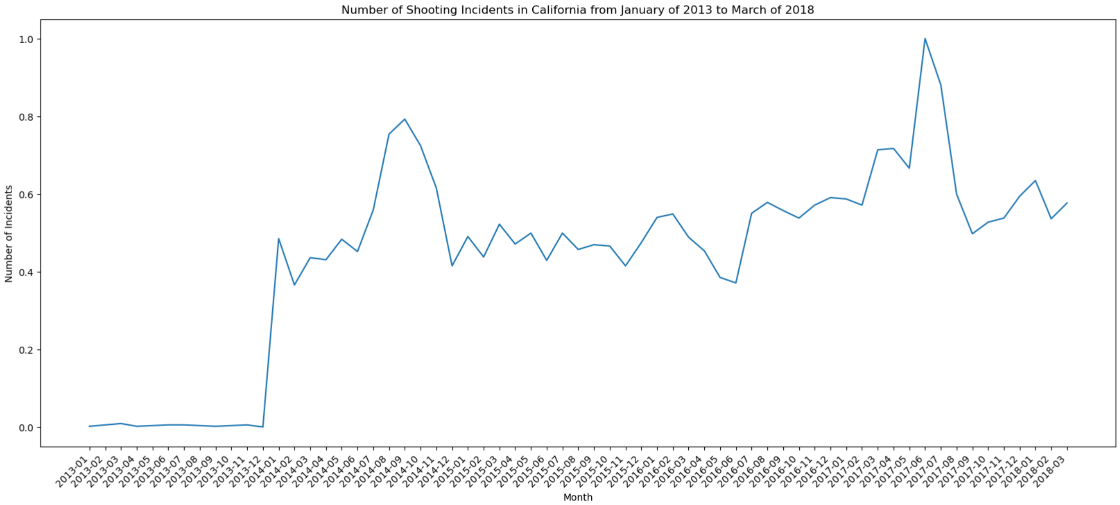
```
[25]: sdf_california = pd.DataFrame.spatial.from_xy(final_df[['longitude','latitude',␣
      ↪'n_killed']] , x_column = 'longitude', y_column='latitude', sr = 4326)
      sdf_california_fl = sdf_california.spatial.to_featurelayer(title ='California␣
      ↪Shooting Point Maps', gis=gis, tags="Final-Project-EDA")
      california_shooting_map = gis.map('California')
      california_shooting_map.add_layer(sdf_california_fl)
      california_shooting_map
```

```
MapView(layout=Layout(height='400px', width='100%'))

<IPython.core.display.HTML object>

<IPython.core.display.HTML object>
```

```
[26]: california['month_year'] = california['date'].apply(lambda x: x[:7])
      cal_inc_by_month = california.groupby('month_year')['incident_id'].count()
```

```
cal_inc_by_month_norm=(cal_inc_by_month - cal_inc_by_month.min()) /␣
 ↪(cal_inc_by_month.max() - cal_inc_by_month.min())
plt.figure(figsize=(20, 8))
plt.xticks(rotation=45, ha="right")
plt.plot(cal_inc_by_month_norm.index, cal_inc_by_month_norm.values)
plt.xlabel("Month")
plt.ylabel("Number of Incidents")
plt.title("Number of Shooting Incidents in California from January of 2013 to␣
 ↪March of 2018")
plt.show()
```

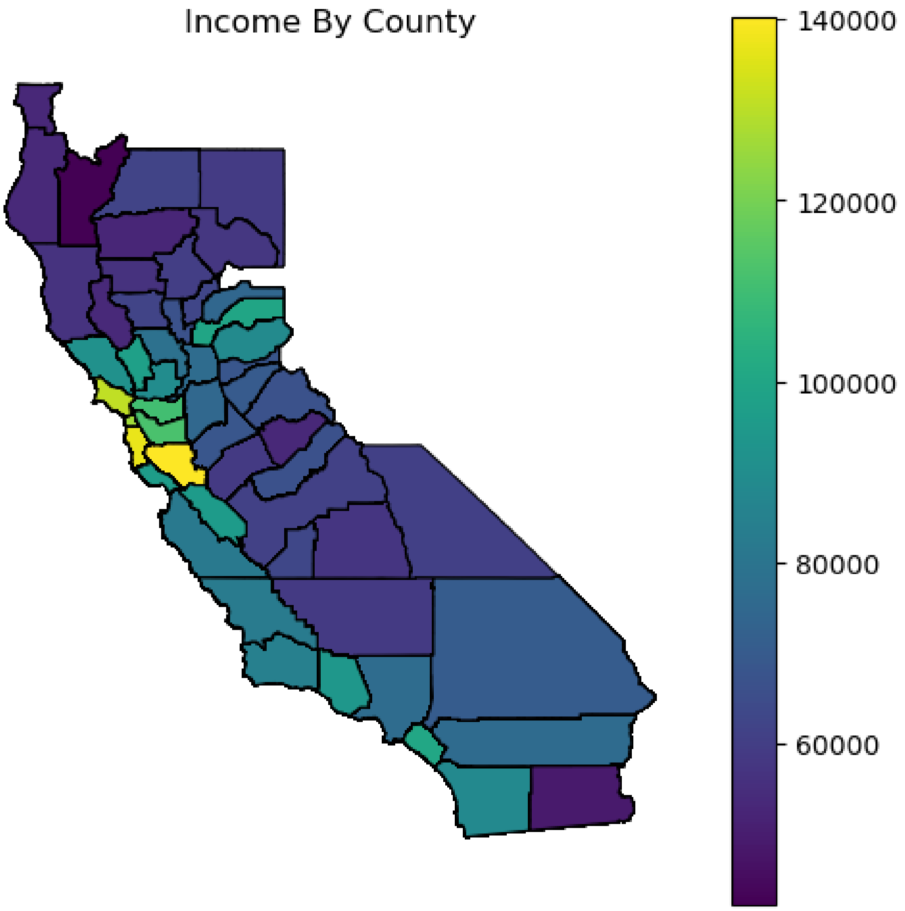
```
/var/folders/dq/1msnq4cn1cb9r_1pcr8n869c0000gn/T/ipykernel_60395/738717121.py:1:
SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead

See the caveats in the documentation: https://pandas.pydata.org/pandas-
docs/stable/user_guide/indexing.html#returning-a-view-versus-a-copy
  california['month_year'] = california['date'].apply(lambda x: x[:7])
```



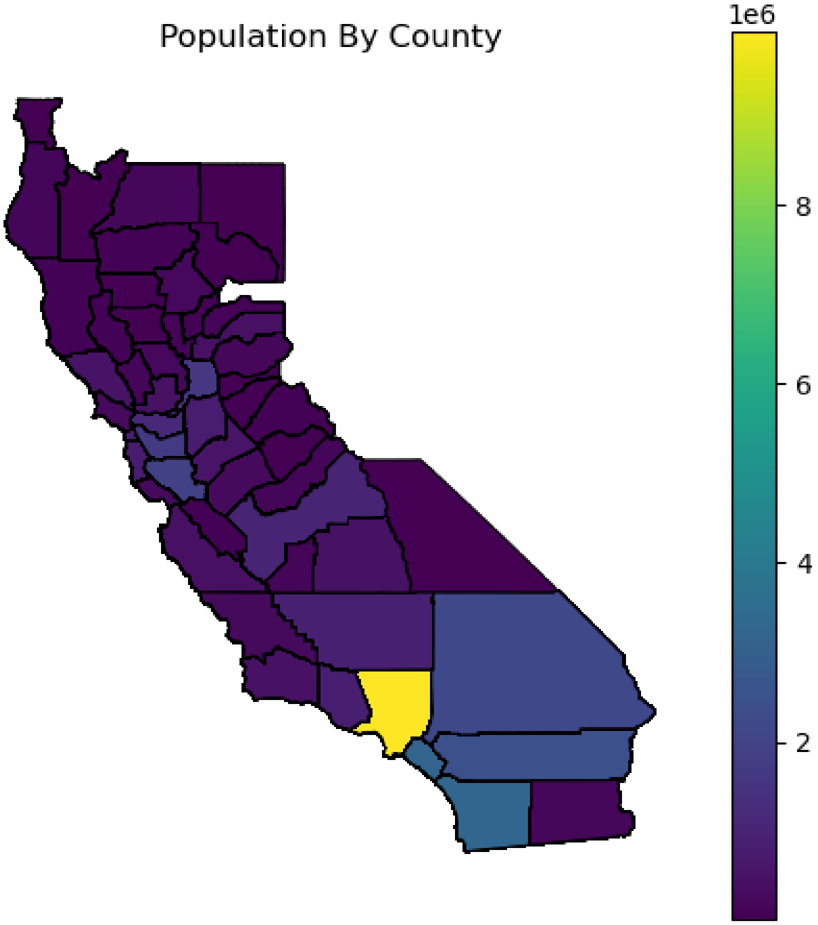
[27]:
```
gdf = gpd.GeoDataFrame(final_df, geometry = 'geometry')
```

[28]:
```
map_income = gdf.plot(
    figsize=(6, 6),
    column="income",
    legend=True,
    edgecolor='black',
    linewidth=0.8
)
```

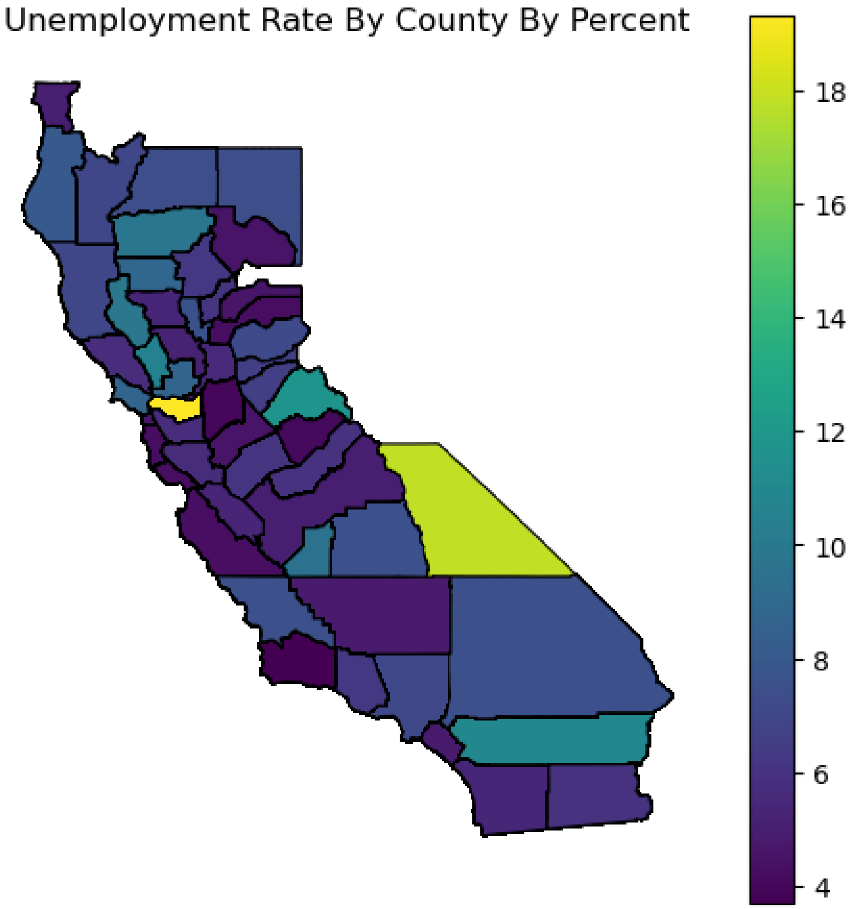
```
map_income.set_axis_off()

plt.title("Income By County")
plt.show()
```



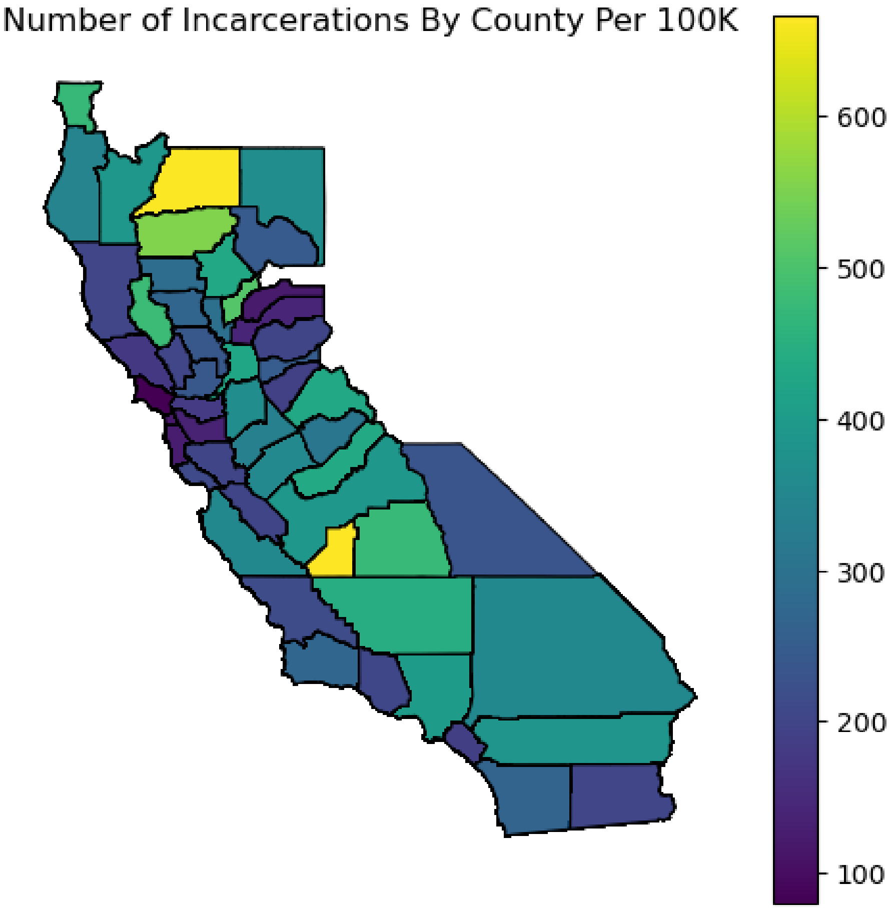
[29]:
```
map_population = gdf.plot(
    figsize=(6, 6),
    column="population",
    legend=True,
    edgecolor='black',
    linewidth=0.8
)

map_population.set_axis_off()

plt.title("Population By County")
plt.show()
```



[30]:
```
map_unemployment = gdf.plot(
    figsize=(6, 6),
    column="unemployment_rate",
    legend=True,
    edgecolor='black',
    linewidth=0.8
)

map_unemployment.set_axis_off()

plt.title("Unemployment Rate By County By Percent")
plt.show()
```

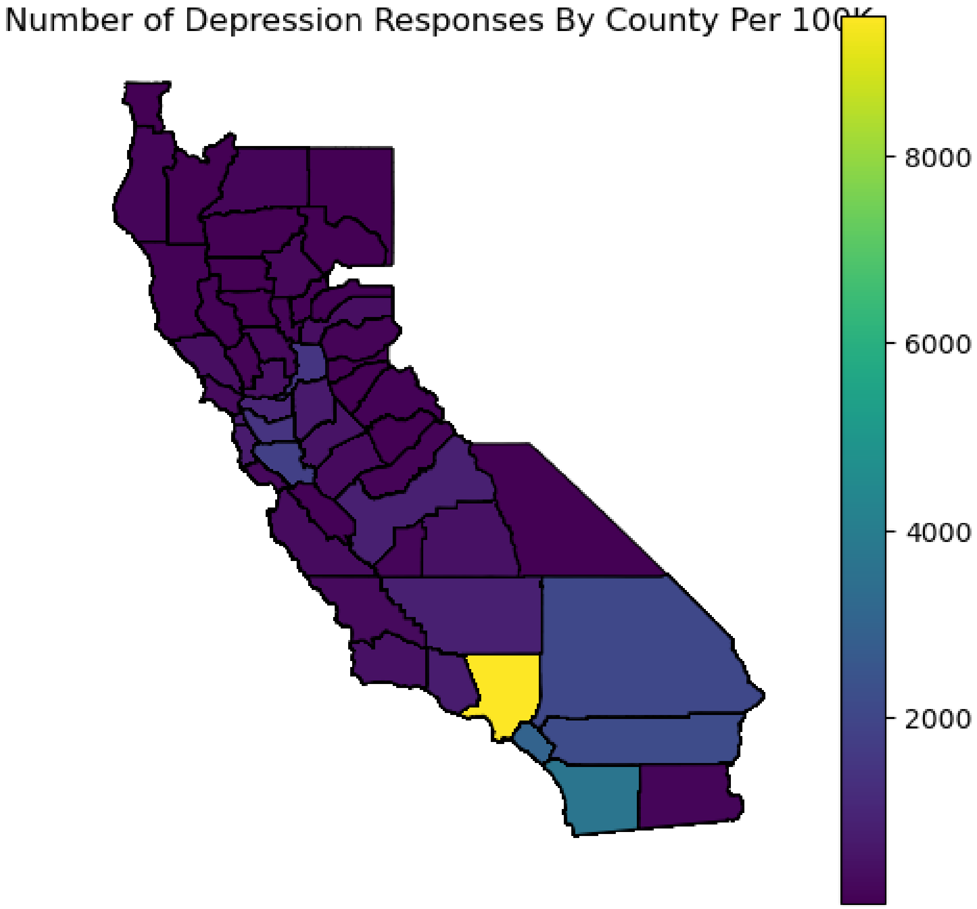


[31]:
```
map_incarceration = gdf.plot(
    figsize=(6, 6),
    column="incarceration_rate",
    legend=True,
    edgecolor='black',
    linewidth=0.8
)

map_incarceration.set_axis_off()

plt.title("Number of Incarcerations By County Per 100K")
plt.show()
```



[32]:

```
map_depression = gdf.plot(
    figsize=(6, 6),
    column="total_depression_responses",
    legend=True,
    edgecolor='black',
    linewidth=0.8
)

map_depression.set_axis_off()

plt.title("Number of Depression Responses By County Per 100K")
plt.show()
```



Number of Depression Responses By County Per 100K

1.8 8. Descriptive statistics for the data

We further analyzed our dataset by looking into the summary statistics of our dataset.

```
[37]: df.describe()
```

```
[37]:        incident_id       n_killed      n_injured       latitude  \
      count  2.317540e+05  231754.000000  231754.000000  231754.000000
      mean   5.516661e+05       0.246719       0.494209      37.546598
      std    2.925579e+05       0.519149       0.731516       5.130763
      min    9.211400e+04       0.000000       0.000000      19.111400
      25%    3.009188e+05       0.000000       0.000000      33.903400
      50%    5.300570e+05       0.000000       0.000000      38.570600
      75%    8.123165e+05       0.000000       1.000000      41.437375
      max    1.083466e+06      50.000000      53.000000      71.336800

                 longitude           year
      count  231754.000000  231754.000000
```

```
mean      -89.338348    2015.684817
std        14.359546       1.225676
min      -171.429000    2013.000000
25%       -94.158725    2015.000000
50%       -86.249600    2016.000000
75%       -80.048625    2017.000000
max        97.433100    2018.000000
```

We first look at our original dataset which contains shootings across the whole country. The most interesting thing to note here is the n_killed and n_injured columns. The average for these variables respectively are 0.25 and 0.49 and the standard deviations are 0.52 and 0.73. This indicates that many of our shooting incidents are smaller scale incidents with some cases even having no injuries or killings. Looking at the max, we see values of 50 and 53.

To have better insight into our area of interest, we take a look at the same summary statistics but on our final dataset which only contains shootings that occurred in California.

[34]:
```
final_df.describe()
```

[34]:
```
       within_police     longitude      latitude   incident_id      n_killed  \
count   15975.000000  15975.000000  15975.000000  1.597500e+04  15975.000000
mean        0.309108   -119.922765     36.110798  5.658317e+05      0.336964
std         0.462140      2.016431      2.090366  3.015250e+05      0.567974
min         0.000000   -124.321000     32.545300  9.216200e+04      0.000000
25%         0.000000   -121.873500     34.043700  2.885465e+05      0.000000
50%         0.000000   -119.779000     36.682000  5.497590e+05      0.000000
75%         1.000000   -118.184000     37.790250  8.470945e+05      1.000000
max         1.000000   -114.589000     41.849900  1.083136e+06     16.000000

          n_injured          year  unemployment_rate         income  \
count  15975.000000  15975.000000       15975.000000   15975.000000
mean       0.469421   2015.732770           6.706930   82785.036682
std        0.790136      1.269648           2.886452   21016.506989
min        0.000000   2013.000000           3.700000   42206.000000
25%        0.000000   2015.000000           5.000000   70287.000000
50%        0.000000   2016.000000           5.700000   76367.000000
75%        1.000000   2017.000000           7.100000   94150.000000
max       19.000000   2018.000000          19.300000  140258.000000

         population  incarceration_rate  total_depression_responses  \
count  1.597500e+04        15975.000000                15975.000000
mean   3.089880e+06          322.083318                 2912.299092
std    3.572012e+06          116.539419                 3450.668612
min    1.588900e+04           80.000000                    5.000000
25%    8.420090e+05          201.000000                  719.000000
50%    1.663823e+06          364.000000                 1418.000000
75%    3.175227e+06          402.000000                 2989.000000
max    9.936690e+06          666.000000                 9502.000000
```

```
       total_victims
count   15975.000000
mean        0.806385
std         0.928998
min         0.000000
25%         0.000000
50%         1.000000
75%         1.000000
max        35.000000
```

Looking at the same columns, we see the average of n_killed and n_injured to be 0.34 and 0.47 and standard deviation to be 0.57 and 0.79. The average number of people killed in a shooting sees a significant increase when we narrow the scope of our research to California indicating that many shooting incidents that occur in California have higher fatality rates in comparison to the rest of the country. When looking at the max of these columns, we see values of 16 and 19 which indicates that the massive shooting incidents which killed 50 and injured 53 did not occur in California but elsewhere. Excluding these outliers though, we can conclude that California shootings seem to be more fatal.

1.9 9. Analysis

After identifying the severity of shooting incidents in California, we were curious as to whether we could accurately predict the number of victims in a shooting given information of features we felt were relevant. We defined the total number of victims in a shooting to be the sum of the number of killed and number of injured in an incident. We treated this as our target variable to predict using a linear regressor. After further consideration these were the features we felt were most relevant for predicting the number of victims: - Year of incident - Month of incident - Average unemployment rate of county the incident occurred in - Average population of county the incident occurred in - Incarceration rate of county the incident occurred in - Total depression responses in county the incident occurred in - Whether or not the shooting occurred within a certain distance from a police station

[40]:
```python
X = final_df[['year', 'month', 'unemployment_rate', 'income', 'population',␣
 ↪'incarceration_rate', 'total_depression_responses', 'within_police']].values
y = final_df[['total_victims']].values
```

Then we did an 80/20 train-test split on our dataset and trained a linear regressor on the training set.

[48]:
```python
train_X, test_X, train_y, test_y = train_test_split(X, y, test_size=0.2)
reg = LinearRegression()
reg.fit(train_X, train_y)
```

[48]:
```
LinearRegression()
```

Upon fitting the model on our training set, we used it to predict the number of victims in incidents in both our training and testing set.

```
[49]: train_pred = reg.predict(train_X)
      test_pred = reg.predict(test_X)
```

```
[50]: train_rmse = np.sqrt(mean_squared_error(train_y, train_pred))
      train_rmse
```

```
[50]: 0.9268653266599014
```

```
[51]: test_rmse = np.sqrt(mean_squared_error(test_y, test_pred))
      test_rmse
```

```
[51]: 0.8831455991661353
```

As you can see, we obtain pretty respectable results with an RMSE of 0.93 on the training set and 0.86 on the testing set. These results show that we can predict the number of victims in a shooting incident with a rough error of about one person.

Next, we wanted to see whether or not we could predict if a shooting incident was a mass killing or not. According to the FBI, they define a mass killing to be a shooting where 3 or more individuals were killed. Using this definition, we create the target variable in our dataset.

```
[55]: final_df['mass killing'] = final_df['n_killed'] >= 3
      final_df.head(2)
```

```
[55]:    within_police  longitude  latitude       county  incident_id        date  \
      0              0   -118.333   33.9090  Los Angeles       460726  2013-01-01
      1              0   -118.131   34.6666  Los Angeles       480407  2013-02-23

              state  n_killed  n_injured  year month  unemployment_rate  income  \
      0  California         1          3  2013    01                7.1   76367
      1  California         0          4  2013    02                7.1   76367

         population  incarceration_rate  total_depression_responses  \
      0     9936690                 402                      9502.0
      1     9936690                 402                      9502.0

                                                  geometry  total_victims  \
      0  MULTIPOLYGON (((-117.66733 34.79317, -117.6672…              4
      1  MULTIPOLYGON (((-117.66733 34.79317, -117.6672…              4

         mass killing
      0         False
      1         False
```

Now that we have our target variable, we can create a prediction task. This task is a classification task where we predict a binary variable (mass killing or not). We choose to train a logistic regressor for this task using the same exact features that were used previously which we deemed were most relevant in regards to shooting severity.

[56]:
```
X = final_df[['year', 'month', 'unemployment_rate', 'income', 'population',␣
 ↪'incarceration_rate', 'total_depression_responses', 'within_police']].values
y = final_df[['mass killing']].values
```

[57]:
```
train_X, test_X, train_y, test_y = train_test_split(X, y, test_size=0.2)
reg = LogisticRegression()
reg.fit(train_X, train_y)
```

```
/Users/rickymiura/anaconda3/lib/python3.11/site-
packages/sklearn/utils/validation.py:1184: DataConversionWarning: A column-
vector y was passed when a 1d array was expected. Please change the shape of y
to (n_samples, ), for example using ravel().
  y = column_or_1d(y, warn=True)
```

[57]:
```
LogisticRegression()
```

Once again, we conduct an 80/20 train-test split on our dataset and train the logistic regressor. After fitting, we can predict mass killing or not on both the training and testing set.

[58]:
```
train_acc = reg.score(train_X, train_y)
train_acc
```

[58]:
```
0.9959311424100157
```

[59]:
```
test_acc = reg.score(test_X, test_y)
test_acc
```

[59]:
```
0.9949921752738654
```

As you can see, we obtained accuracies of 99 percent on both sets of the data. The model seems to be overfitting so we looked further into our data to see any potential biases.

[60]:
```
final_df['mass killing'].value_counts()
```

[60]:
```
mass killing
False    15907
True        68
Name: count, dtype: int64
```

We identified that we have a very big class imbalance with 15907 incidents not being mass killings and only 68 that were. To balance the dataset, we randomly sample the non-mass killing incidents so we have the same number of both.

[61]:
```
not_mass_killing = final_df[final_df['mass killing'] == False]
mass_killing = final_df[final_df['mass killing'] == True]
random_not = not_mass_killing.sample(mass_killing.shape[0])
balanced = pd.concat([mass_killing, random_not])
balanced['mass killing'].value_counts()
```

```
[61]: mass killing
      True     68
      False    68
      Name: count, dtype: int64
```

Now that we have an equal number of incidents of both classes, we can retrain the model and obtain different results.

```
[62]: X = balanced[['year', 'month', 'unemployment_rate', 'income', 'population',␣
        ↪'incarceration_rate', 'total_depression_responses', 'within_police']].values
      y = balanced[['mass killing']].values
```

```
[63]: train_X, test_X, train_y, test_y = train_test_split(X, y, test_size=0.2)
```

```
[64]: reg = LogisticRegression()
      reg.fit(train_X, train_y)
```

```
/Users/rickymiura/anaconda3/lib/python3.11/site-
packages/sklearn/utils/validation.py:1184: DataConversionWarning: A column-
vector y was passed when a 1d array was expected. Please change the shape of y
to (n_samples, ), for example using ravel().
  y = column_or_1d(y, warn=True)
```

```
[64]: LogisticRegression()
```

```
[65]: train_acc = reg.score(train_X, train_y)
      train_acc
```

```
[65]: 0.5833333333333334
```

```
[66]: test_acc = reg.score(test_X, test_y)
      test_acc
```

```
[66]: 0.6071428571428571
```

After balancing the dataset, we got an accuracy of 58 percent on the training set and 60.7 percent on the testing set. By using this method, we overcame issues of overfitting while still obtaining respectable results from our model.

1.10 10. Summary of Products and Results

By gathering, cleaning and analyzing the data, our project created successful analytic models and revealed findings about gun violence.

Our exploratory data analysis compared California's rate of gun violence to other states, specifically with a focus on severity (number of fatalities and injuries). We could identify a significant diversity in the presence of shootings and the incident characteristics across the different counties within California, leading us to investigate the influence of local factors such as demographic data. Through geospatial visualization of the demographic data, we showcased the distribution of income, population, unemployment rates, incarceration rates, and depression responses across various counties in California. The income choropleth indicated that income was generally higher in metropolitan areas such as the Bay Area or southern California, with the Bay Area having a disproportionately high income when compared to other counties. The population choropleth indicated that population was again higher in metropolitan areas but now with Los Angeles county being disproportionately high when compared to other counties. The unemployment choropleth indicated that there was no clear trend in unemployment rate by county. The incarceration choropleth indicated that there was no visible trend in incarceration rate by county except for an exceptionally low rate in the Bay Area, which had an noticeably low rate of incarceration . The mental health choropleth showed no pattern for California counties except for Los Angeles county which had an extremely high rate of depression, possibly due to their disproportionally high population.

Important findings of our analysis revealed that demographic characteristics, including income levels, unemployment rates, population, incarceration rates, and mental health statistics (indicated by depression responses), play some kind of role in the presence of shootings. These factors were the only pieces of data that went into the predicting the severity of shootings in our first model and our model was able to achieve a high prediction accuracy. Via our linear regression model, we could predict the number of victims in a shooting incident with a reasonable RMSE of 0.93 indicating the influence of the socioeconomic and environmental variables allowed our model to predict within a less than 1 person margin of error. However the logistic regression model, which aimed to predicting if a shooting qualifies as a mass killing or not, initially showed high accuracy due to class imbalance. Once we addressed this issue by balancing the dataset, the model demonstrated significantly worse performance with an accuracy of 68% indicating that while the features might hold predictive influence for one aspect of gun violence (severity), they do not necessarily hold predictive weight for all aspects of gun violence.

1.11 11. Discussion

Our findings align with the existing literature, referenced earlier, concerning socioeconomic trends, patterns in mental health, and gun violence.

1) In our project the trend of higher levels of shooting occurring in areas of lower income and higher unemployment rates agrees with previous literature, such as the BMC Public Health piece, which highlights economic disparities as motivating factors in the prevalence of gun violence. Our analysis also echoes the importance of mental health issues by sharing the link between shootings and depression rate, which was also done in the research by Columbia University. Our research contributes to the understanding of this problem by focusing on California specifically (the state with the highest rate of mass shootings) to investigate the characteristics of shootings and conduct analysis on a state level, rather than a national level since it could reveal different findings. It also integrates geospatial insights into findings which something that is not present in existing literature.

2) We came across multiple points of trade-off, especially when sourcing data and the determining spatial parameters such as buffer distances for proximity to police stations. The decision to focus on county-level socioeconomic data was helpful for analysis because the data was more widely available but it may have masked finer-grained patterns that could show up in city-level data. County level data often encompasses a diverse array of individual cities and as such it can overgeneralize the trends of an area. Additionally, the choice of the buffer as 1 mile as being “near” a police station reflected the balance between precision and the practicality of representing geographical relation-

ships. If we changed the buffer the be higher (5 miles), the proportion of "near" shootings tripled, making it an unhelpful feature as nearly every shooting was classified as "near". Additionally, the choice of our socio-economic features was driven by the relationship we thought each feature had to public health and criminal justice meaning these factors reflect our own biases. We also chose to implement relatively simple models for prediction such as linear and logistic regression, rather than more complex models such a decision trees. This was because we wanted to investigate the existence of a relationship between these patterns and we wanted to provide straightforward interpretations of the effects of the variables on the outcome.

1.12 12. Conclusions and Future Work

Our exploration into the socioeconomic and environmental factors that affect shootings within California produced meaningful and insightful results. Through utilizing spatial data and demographic data we have gained a better understanding of what factors could correlate with gun violence specifically income level, rates of depression, rates of incarceration and incidents of gun violence. Our project sheds a light on how geographical and socioeconomic differences can increase the risk and severity of shootings, putting certain communities at risk. We are choosing to say we did manage to answer our question successfully based on the success of our prediction models and the trends visualized in our data analysis. While our project does successfully answer the question, there are still limitations in our methodology and data. The complexity of gun violence and the questionable ability of county-wide sociodemographic data to accurately reflect a community mean that our conclusions can serve as a starting off point for answering such questions but not the firm answer.

In the future this methodology can be used to explore other states or nations to see whether these trends are unique or generalizable. Additionally, it could also be used to explore the same nature of data over time, rather than a five year period. This strain of research could be scaled up to operate on a national level or down to a more granular level, such as city by city, as both types of exploration could provide valuable insights. Research such as this could be useful for local government, law enforcement and community leaders. The insights taken from this investigation could help these parties better prevent and prepare for gun violence and could give them an understanding of what might go into these incidents. It could also help all of these parties with resource allocation, whether that be the placement of law enforcement offices, or the amount of mental health resources distributed to each county.