DIC PROJECT

Name: Adikavya Gupta

UB person no: 50419660

Title: United States Mass Shooting Analysis

TASK 1 <u>**Title, Problem statement, and questions**</u>

Title

United States Mass Shooting Analysis

Problem Statement

Analysis of mass shooting data based on time, location, shooter demographics including mental health, and impact of gun laws on these incidents.

By looking at the datasets of Mass shootings the **questions** I aim to answer or understand are

1. *Are there certain states more susceptible to gun violence?*
2. *Are there certain cities more susceptible to shootings?*
3. *What are the different locations for these incidents?*
4. *Are there years in which the incidents are more?*
5. *Is there a particular month when incidents peak?*
6. *What is the specific demographic profile for the shooters (Age Group/Race/Gender)?*
7. *Will the implementation of stricter gun laws have an impact on these incidents?*
8. *Does the shooter have any prior mental health issues?*

Background

Every year thousands of people lose their lives or are gravely injured due to gun violence in the United States. Mass shootings have detrimental effects on the health of the people who witness them, those who live in the communities surrounding them, and those who identify with the demographic groups targeted in them. Emerging literature shows that mass shootings increase depression and other mental health disorders among teenagers and adults, worsen infant health and reduce overall community and emotional well-being. It has become extremely urgent to solve this issue since it has such a huge impact on society as a whole. The United States is recognized as a nation of immigrants, and thousands of individuals move here each year to start new lives. These incidents have an impact on individuals all throughout the world, not just in one nation.

Why this problem?

The reason for choosing this problem is recently an incident took place in Buffalo (TOPS) which was reported as a mass shooting. Fortunately, no one I knew was present there, but it got me wondering about the graveness of this problem and about the thousands of innocent people losing their lives yearly due to gun violence.

Why is this important?

Analysis of this problem can make a difference in people's lives. If we can present a certain concrete analysis to understand the reasons for such incidents occurrence, maybe the concerned authorities/government bodies can implement policies that prevent and mitigate the problem thus helping save the lives of thousands of innocent people and help shooters not to indulge in such heinous crime.

TASK 2 Dataset

Source: U S Mass Shooting Data (Kaggle)

Timeframe: 1982-2023

Format: .csv file

<u>Why this Dataset?</u>

The reasons for choosing this dataset are

1. It is a very comprehensive dataset.
2. Although many datasets of similar type exist most of them lack shooter demographics.
3. It is quite extensive and detailed.
4. The dataset has listed the original reporting source for each incident making it quite reliable.

TASK 3 **<u>Cleaning the data and Preliminary Analysis (Understanding the Data)</u>**

TASK 3A Cleaning the Data

The following steps were taken to clean the data.

Step 1: After importing the data from .csv file, I implemented df. shape to get the shape of the data.

```
In [62]: df.shape
Out[62]: (141, 19)
```

Step 2: Next, to understand the kind of data in the columns I used df.info to see the data in the columns.

```
In [63]: df.info
Out[63]: <bound method DataFrame.info of                                    case                  location  \
         0      Nashville religious school shooting       Nashville, Tennessee
         1       Michigan State University shooting      East Lansing, Michigan
         2             Half Moon Bay spree shooting   Half Moon Bay, California
         3            LA dance studio mass shooting   Monterey Park, California
         4                 Virginia Walmart shooting        Chesapeake, Virginia
         ..                                     ...                        ...
         136         Shopping centers spree killings           Palm Bay, Florida
         137   United States Postal Service shooting            Edmond, Oklahoma
         138          San Ysidro McDonald's massacre     San Ysidro, California
         139               Dallas nightclub shooting               Dallas, Texas
         140                   Welding shop shooting              Miami, Florida

                    date                                            summary  \
         0     3/27/2023  Audrey Hale, 28, who was a former student at t...
         1     2/13/2023  Anthony D. McRae, 43, opened fire at Berkey Ha...
         2     1/23/2023  Chunli Zhao, 67, suspected of carrying out the...
         3     1/21/2023  Huu Can Tran, 72, fled the scene in a white va...
         4    11/22/2022  Andre Bing, 31, who worked as a supervisor at ...
         ..          ...                                                ...
         136   4/23/1987  Retired librarian William Cruse, 59, was paran...
         137   8/20/1986  Postal worker Patrick Sherrill, 44, opened fir...
         138   7/18/1984  James Oliver Huberty, 41, opened fire in a McD...
         139   6/29/1984  Abdelkrim Belachheb, 39, opened fire at an ups...
         140   8/20/1982  Junior high school teacher Carl Robert Brown, ...

              fatalities  injured  total_victims location.1 age_of_shooter  \
         0             6        1              6     School             28
         1             3        5              8     School             43
         2             7        1              8       work             67
         3            11       10             21      Other             72
         4             6        6             12       work             31
         ..          ...      ...            ...        ...            ...
         136           6       14             20      Other             59
         137          15        6             21       work             44
```

Step 3: To identify data types I implemented df.dtypes.

In [64]: df.dtypes

```
Out[64]: case                                 object
location                             object
date                                 object
summary                              object
fatalities                            int64
injured                               int64
total_victims                         int64
location.1                           object
age_of_shooter                        int64
prior_signs_mental_health_issues     object
mental_health_details                object
weapons_obtained_legally             object
where_obtained                       object
weapon_type                          object
weapon_details                       object
race                                 object
gender                               object
type                                 object
year                                  int64
dtype: object
```

Step 4: I converted the object datatypes into strings and integers to work on them later.

```
In [65]: df['location'] = df['location'].astype('string')
         df['summary'] = df['summary'].astype('string')
         df['location.1'] = df['location.1'].astype('string')
         df['where_obtained'] = df['where_obtained'].astype('string')
         df['weapon_type']=df['weapon_type'].astype('string')
         df['weapon_details']=df['weapon_details'].astype('string')
         df['race']=df['race'].astype('string')
         df['gender']=df['gender'].astype('string')
         df['type']=df['type'].astype('string')
```

```
In [66]: df.dtypes
```

```
Out[66]: case                                object
         location                            string
         date                                object
         summary                             string
         fatalities                           int64
         injured                              int64
         total_victims                        int64
         location.1                          string
         age_of_shooter                       int64
         prior_signs_mental_health_issues    object
         mental_health_details               object
         weapons_obtained_legally            object
         where_obtained                      string
         weapon_type                         string
         weapon_details                      string
         race                                string
         gender                              string
         type                                string
         year                                 int64
         dtype: object
```

Step 5: Using drop.duplicates I removed any duplicates in the data. Fortunately, there were no duplicates in the dataset.

In [58]:
```
df.drop_duplicates()
```

Out[58]:

	case	location	date	summary	fatalities	injured	total_victims	location.1	age_of_shooter	prior_signs_mental_health_issues	mental_healt
0	Nashville religious school shooting	Nashville, Tennessee	3/27/2023	Audrey Hale, 28, who was a former student at t...	6	1	6	School	28	NaN	
1	Michigan State University shooting	East Lansing, Michigan	2/13/2023	Anthony D. McRae, 43, opened fire at Berkey Ha...	3	5	8	School	43	NaN	
2	Half Moon Bay spree shooting	Half Moon Bay, California	1/23/2023	Chunli Zhao, 67, suspected of carrying out the...	7	1	8	work	67	NaN	
3	LA dance studio mass shooting	Monterey Park, California	1/21/2023	Huu Can Tran, 72, fled the scene in a white va...	11	10	21	Other	72	yes	According Times, enfo
4	Virginia Walmart shooting	Chesapeake, Virginia	11/22/2022	Andre Bing, 31, who worked as a supervisor at ...	6	6	12	work	31	NaN	

Step 6: For preliminary statistical analysis I used .describe() function and obtained min, max, std etc.

In [57]:
```
df.describe()
```

Out[57]:

	fatalities	injured	total_victims	age_of_shooter	year
count	141.000000	141.000000	141.000000	141.000000	141.000000
mean	7.808511	11.205674	19.007092	34.106383	2010.382979
std	7.463162	46.579505	51.747532	13.165269	10.796600
min	3.000000	0.000000	3.000000	11.000000	1982.000000
25%	4.000000	1.000000	6.000000	23.000000	2005.000000
50%	6.000000	3.000000	10.000000	33.000000	2014.000000
75%	8.000000	10.000000	17.000000	43.000000	2018.000000
max	58.000000	546.000000	604.000000	72.000000	2023.000000

Step 7: I found the missing values in my data. The columns that have null values are race, prior_signs_mental_health_issues, and weapon_details. I did not try to fill in

the values using any average or any machine learning algorithm as the accuracy of finding the missing values is extremely low in this.

```
In [60]: df.isna().sum()

Out[60]: case                              0
         location                          0
         date                              0
         summary                           0
         fatalities                        0
         injured                           0
         total_victims                     0
         location.1                        0
         age_of_shooter                    0
         prior_signs_mental_health_issues 28
         mental_health_details             0
         weapons_obtained_legally          0
         where_obtained                    0
         weapon_type                       0
         weapon_details                    1
         race                             13
         gender                            0
         type                              0
         year                              0
         dtype: int64
```

Step 8: I split the date into 3 additional columns (Year, Month, and Date) for detailed time analysis.

Step 9: I split the location data into states and cities to be able to analyze it in depth.

```
In [67]: df[['Month','Day','year']] = df['date'].str.split('/', expand=True)
         df[['City','State']] = df['location'].str.split(',',1, expand=True)
         df.head(2)
```

Out[67]:

	case	location	date	summary	fatalities	injured	total_victims	location.1	age_of_shooter	prior_signs_mental_health_issues	...	weapon_type	w
0	Nashville religious school shooting	Nashville, Tennessee	3/27/2023	Audrey Hale, 28, who was a former student at t...	6	1	6	School	28	NaN	...	semiautomatic rifle, semiautomatic handgun	
1	Michigan State University shooting	East Lansing, Michigan	2/13/2023	Anthony D. McRae, 43, opened fire at Berkey Ha...	3	5	8	School	43	NaN	...	semiautomatic handguns	

2 rows × 23 columns

Step 10: I dropped columns that would not be required for univariate/multivariate analysis.

```
df.drop(['summary','mental_health_details','case'], axis=1)
```

	location	date	fatalities	injured	total_victims	location.1	age_of_shooter	prior_signs_mental_health_issues	weapons_obtained_legally	where_obta
0	Nashville, Tennessee	3/27/2023	6	1	6	School	28	NaN	unknown	
1	East Lansing, Michigan	2/13/2023	3	5	8	School	43	NaN	yes	
2	Half Moon Bay, California	1/23/2023	7	1	8	work	67	NaN	unknown	
3	Monterey Park, California	1/21/2023	11	10	21	Other	72	yes	unknown	
4	Chesapeake, Virginia	11/22/2022	6	6	12	work	31	NaN	unknown	
...	...	...	...	...	...	...	...	...	...	
136	Palm Bay, Florida	4/23/1987	6	14	20	Other	59	Yes	Yes	Gun sto Norwood, C The C Tradin
137	Edmond, Oklahoma	8/20/1986	15	6	21	work	44	Unclear	Yes	Issue Oklah National Gu where Sh

TASK 3B Understanding the Data (Univariate & Multivariate Analysis):

Step 1: I tried to arrange the total number of people injured, killed, and total victims in descending order Grouped by State.

```
In [69]: df.groupby('State')[['total_victims', 'injured', 'fatalities'
```

Out[69]:

State	total_victims	injured	fatalities
Nevada	616	553	63
California	346	171	175
Texas	334	183	151
Florida	235	109	126
Colorado	182	129	53
Illinois	102	77	25
Virginia	88	35	53
New York	68	28	40
Washington	65	28	37
Ohio	56	36	20
Oregon	47	34	13
Connecticut	46	5	41
Pennsylvania	40	13	27
Wisconsin	37	9	28
Michigan	37	19	18

Hypothesis: From the above analysis we can say that Nevada is the worst affected state in the US. Another interesting observation that can be made is based on gun policies. Texas has the most lenient gun laws in the country and while California has the strictest policies for obtaining weapons, both have similar numbers of victims.

Step 2: I tried to arrange the total number of people injured, killed, and total victims in descending order Grouped by Year.

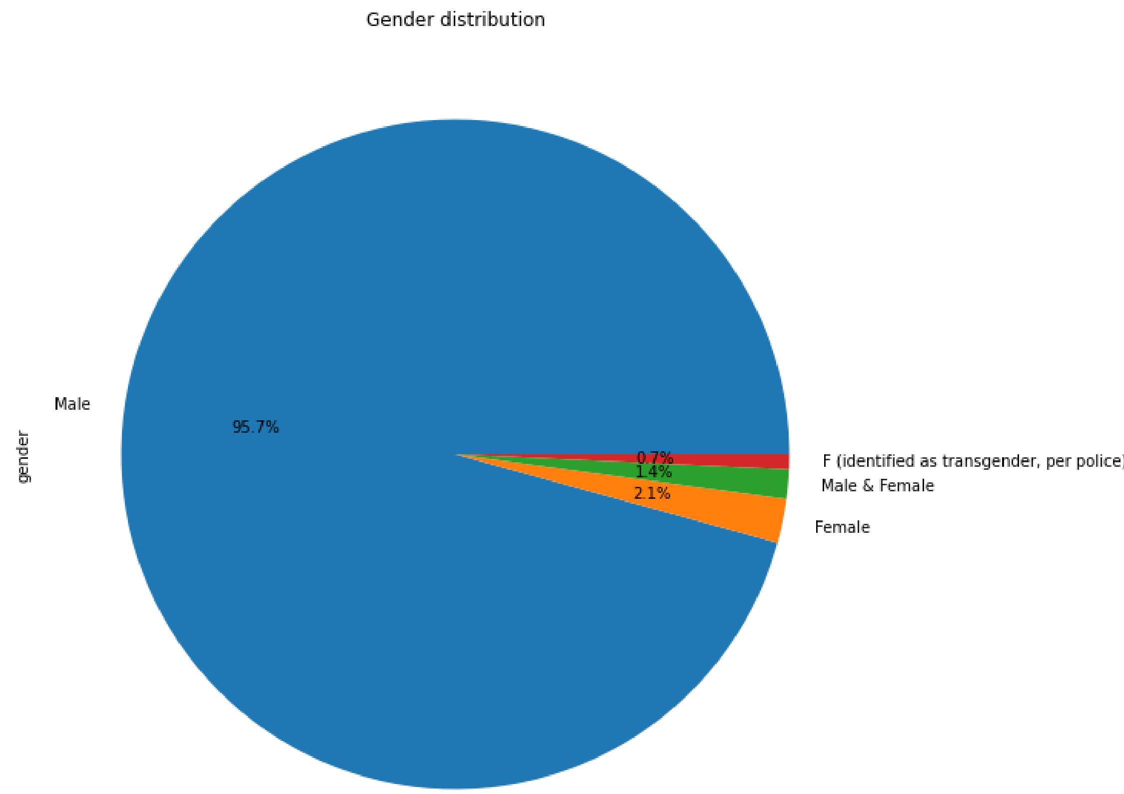
```
In [71]: df.groupby('year')[['total_victims', 'injured', 'fatal
```

Out[71]:

year	total_victims	injured	fatalities
2017	704	587	117
2019	185	112	73
2022	178	104	74
2016	154	83	71
2012	151	80	71
2018	150	70	80
2015	89	43	46
1999	89	47	42
2007	85	32	53
2009	78	39	39
1991	61	26	35
2021	59	16	43
1993	57	34	23
1989	56	41	15
1998	50	36	14

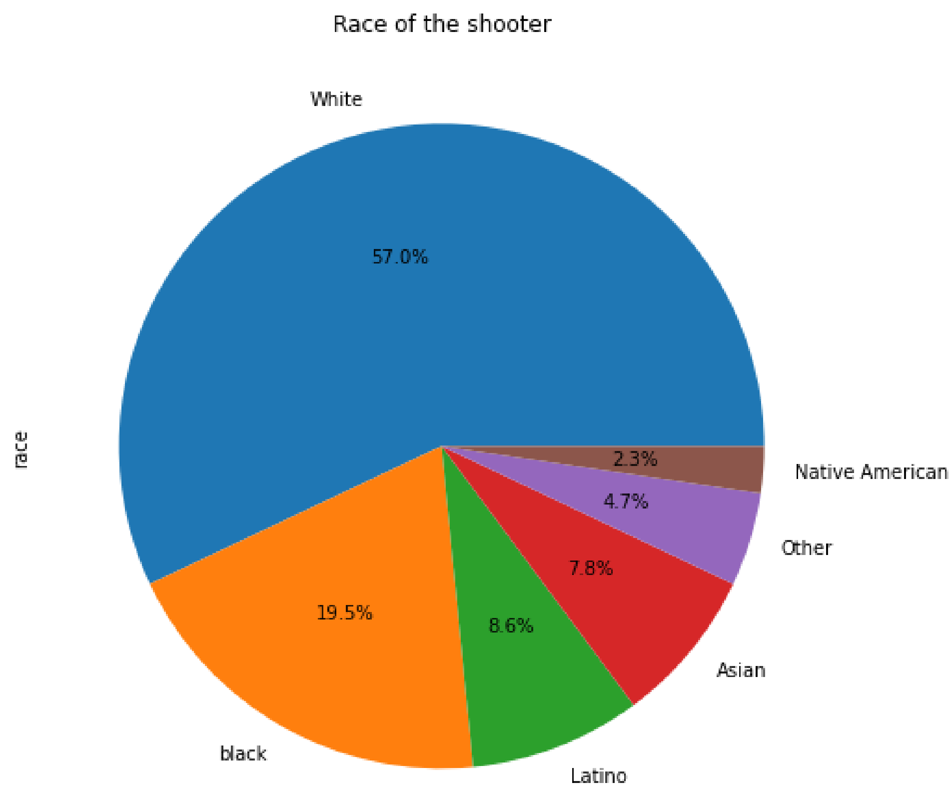
Hypothesis: Most incidents have happened after the 2010s with 2017 being an outlier in the decade. Upon research, I could not find any particular reason why 2017 was so high but 2018 saw a dip due to changes in laws after the shocking statistics of 2017.

Step 3: Next, to understand shooter demographics I tried finding the percentage of shooters that are male/female.



<u>Hypothesis</u>: According to the statistics, 95.7% of shooters are male giving them a majority by a large margin.

Step 4: Continuing shooter demographics, I tried finding how different races fared.



Hypothesis: More than 50% of shooters are white and another major 19.5% are black.

Step 5: Next I created a simple function to classify the age of the shooters into certain age groups to see which group had the most shooters.

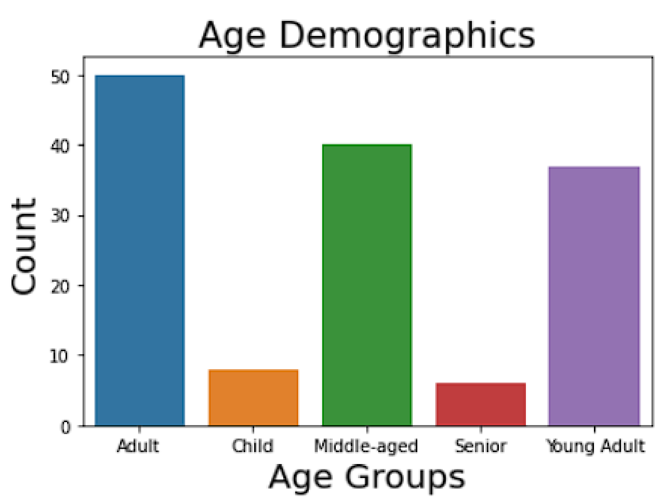
In [118]:
```
def divide_ages_into_groups(df, column_name):
    # Define the age ranges and corresponding labels
    age_ranges = [
        (0, 17, 'Child'),
        (18, 25, 'Young Adult'),
        (26, 40, 'Adult'),
        (41, 60, 'Middle-aged'),
        (61, float('inf'), 'Senior')
    ]

    # Create a new column to store the age groups
    df['Age Group'] = ''

    # Iterate over each row in the DataFrame
    for index, row in df.iterrows():
        age = row[column_name]

        # Check the age against each age range
        for range_start, range_end, group_label in age_ranges:
            if range_start <= age <= range_end:
                df.at[index, 'Age Group'] = group_label
                break

    return df
```

In [119]:
```
df = divide_ages_into_groups(df, 'age_of_shooter')
```

Results

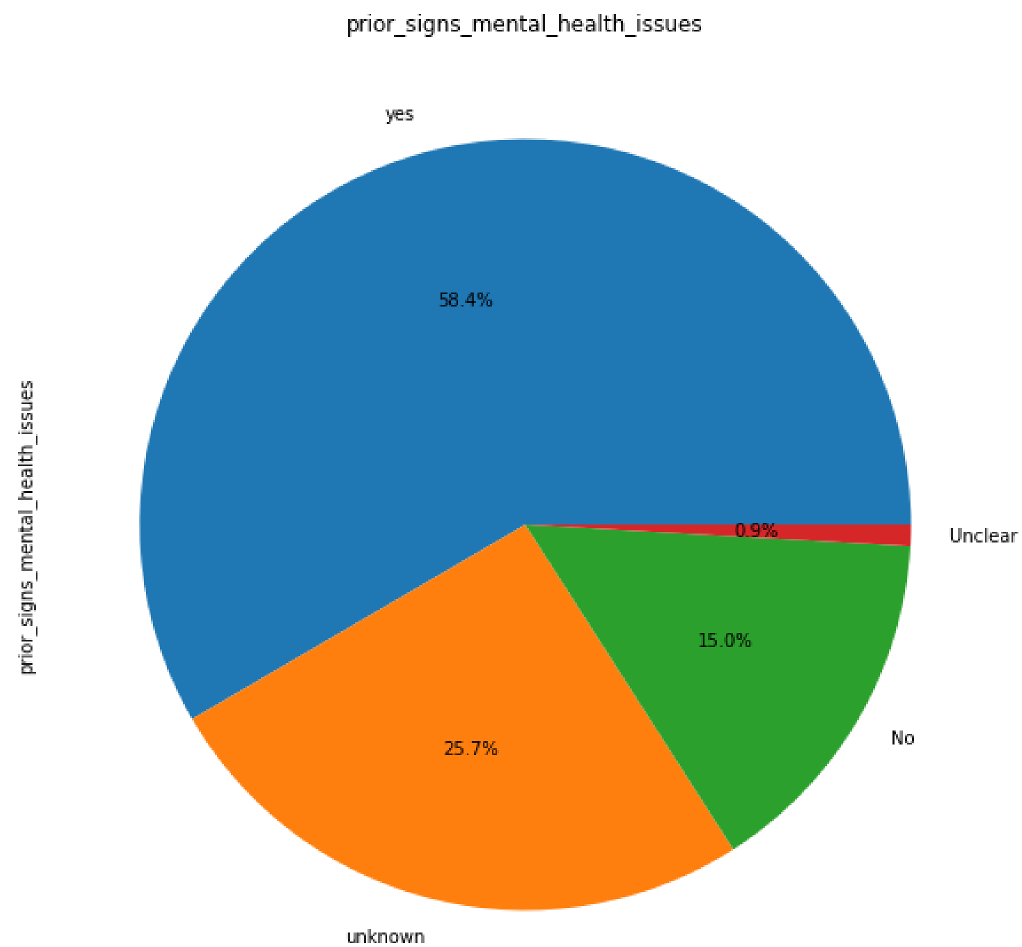
In [136]:
```
Ages = df["Age Group"].sort_values(ascending=True)
sns.countplot(x=Ages, data=df)
plt.xlabel("Age Groups", fontsize=20)
plt.ylabel("Count", fontsize=20)
plt.title('Age Demographics', fontsize = 21)
```

Out[136]:
```
Text(0.5, 1.0, 'Age Demographics')
```



Hypothesis: Although it is shocking that there are a few cases where the age is less than 18 years most are from the age group (30 – 45) and the second most common age group is middle-aged (45-60).

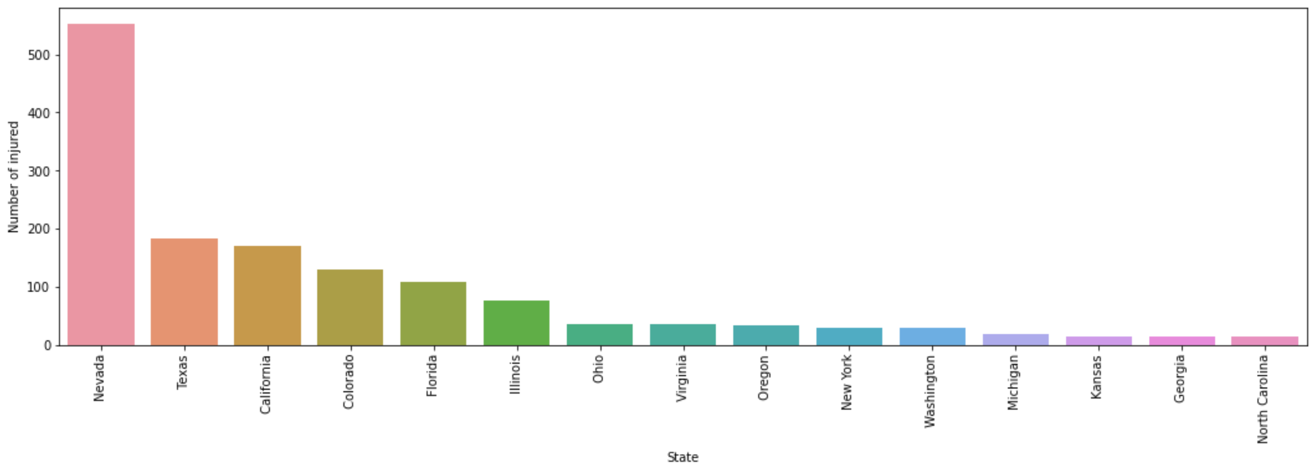
Step 6: Furthermore, an important point of consideration is to consider the mental health of the shooter prior to the incident.



Hypothesis: 58.4 % of people who are shooters have shown signs of mental health issues. A lot of the data in this field is unknown. Based on the current trend may be a higher percentage of people may have had mental health issues which highlights the importance of getting the diagnosis and right treatment at the right time. Awareness of this can help reduce these incidents leading to a safer society for all.

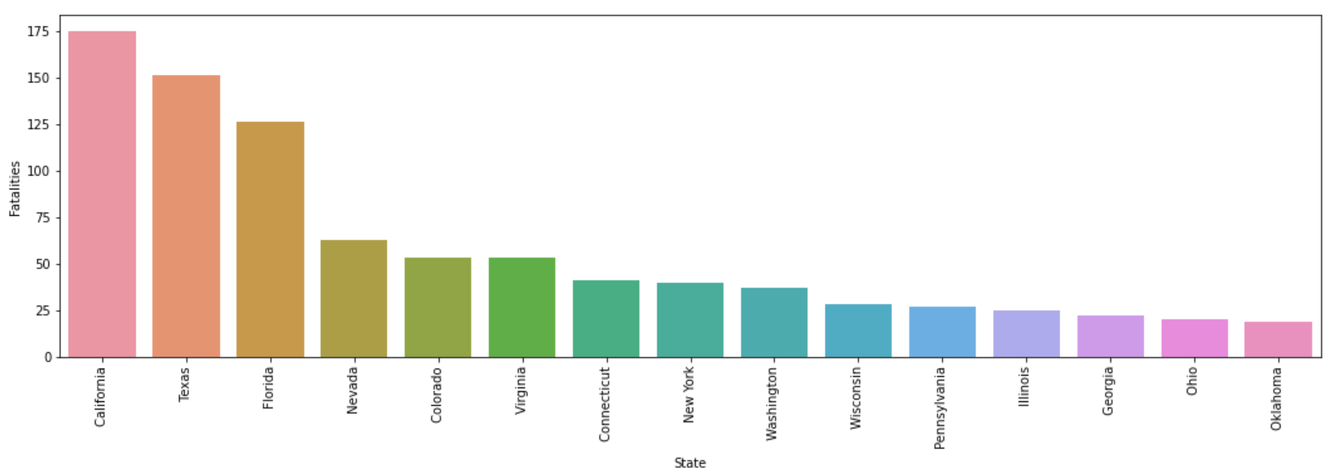
This is one area where we as citizens can actively help remove the stigma around mental health.

Step 7: Next, we try to get information about most Injuries state-wise.



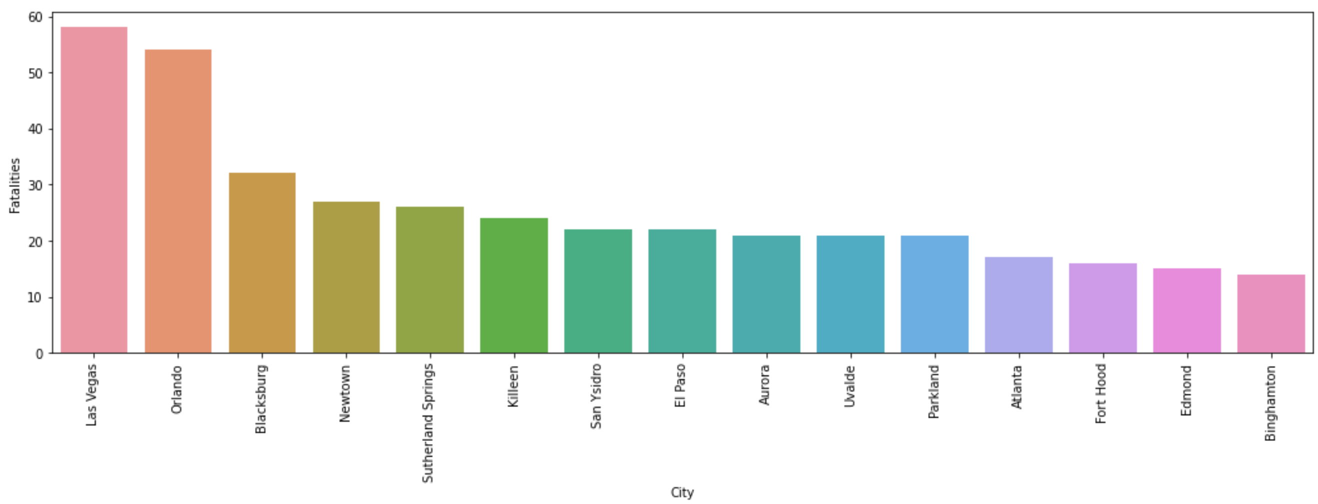
<u>Hypothesis</u>: Nevada had an incident where a mass shooting may have led to some sort of stampede/chaos that may have resulted in so many injured. Or it could have led to the collapse of some place leading to high injuries.

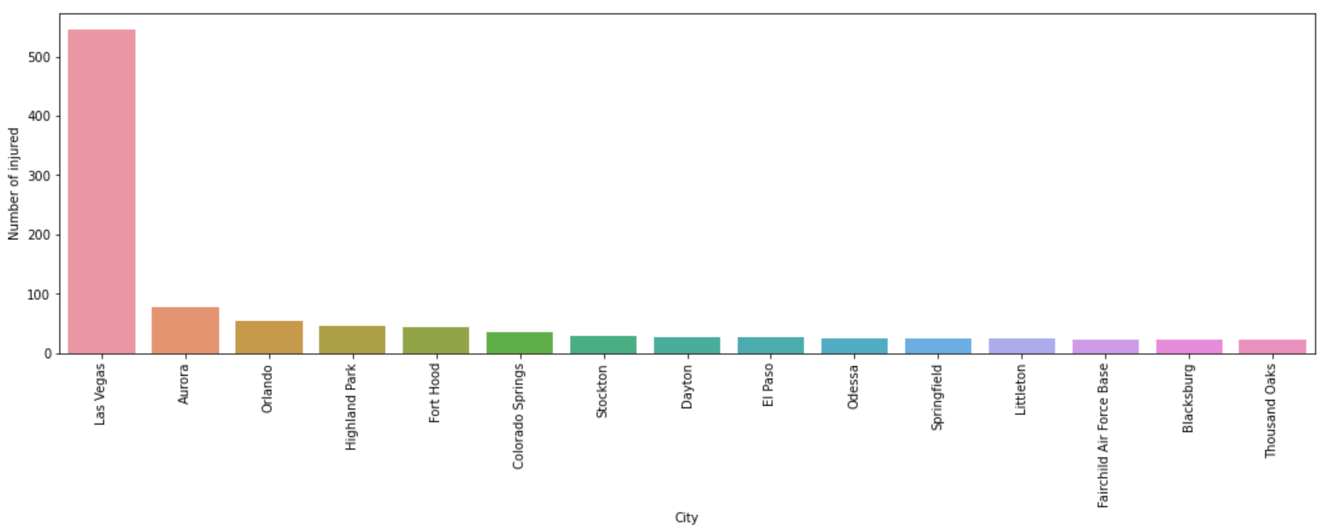
Step 8: Next, we try to get information about most fatalities state-wise.



Hypothesis: California is the most dangerous state with maximum fatalities due to these shootings. Texas is a close second.

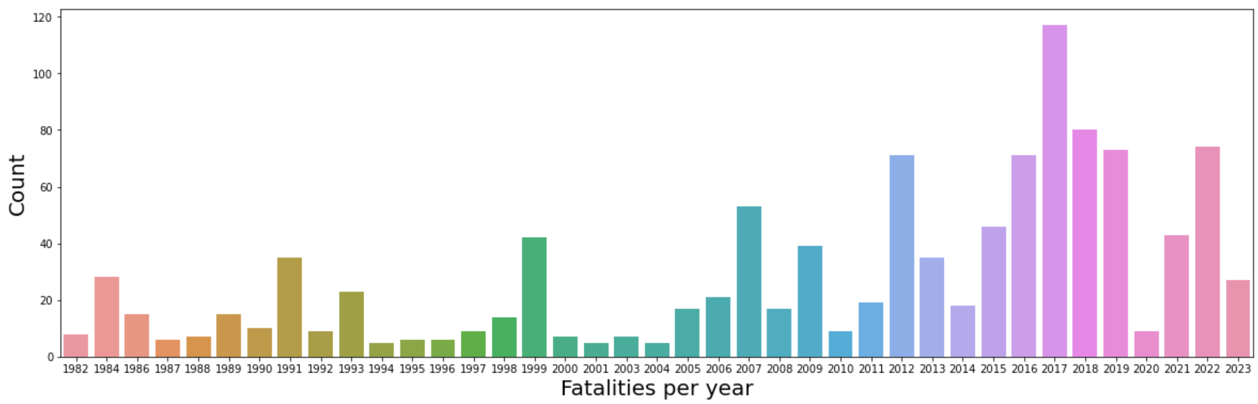
Step 9: If we wish to delve deeper to find the most dangerous cities we can do that as well.

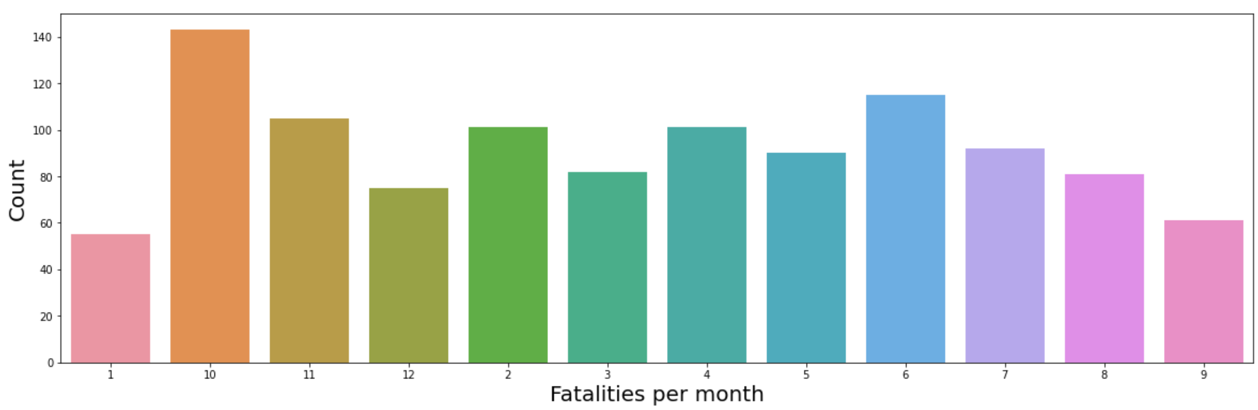




Hypothesis: Las Vegas is one of the most dangerous cities with maximum fatality and injury rates. A more data-filled analysis of this was given in Steps 1 and 2. This is for the user to easily grasp it and not get overwhelmed by sheer numbers.

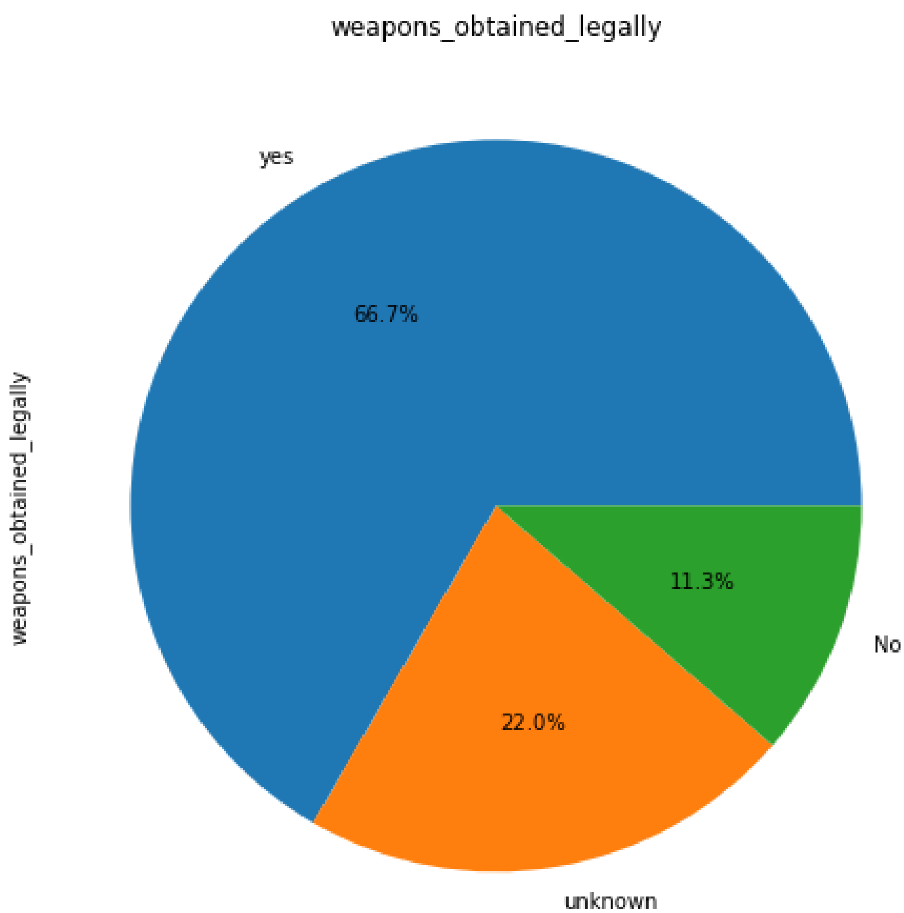
Step 10: To provide the user with a better visualization of the number of cases that took place over the years, we can plot the data in a bar chart. I also plotted the incidents month-wise to see if there are certain times in a year when the frequency of these shootings increases.





<u>Hypothesis</u>: October and June are the months with the maximum of these shootings with October being the top. One hypothesis that can be made is maybe due to Halloween it gets easier to illegally carry weapons in the month of October. 2020 has a drop in cases as compared to the years before and after. This may be due to the outbreak of the coronavirus worldwide.

Step 11: In this step, I tried finding out what percentage of the weapons were obtained legally vs illegally.

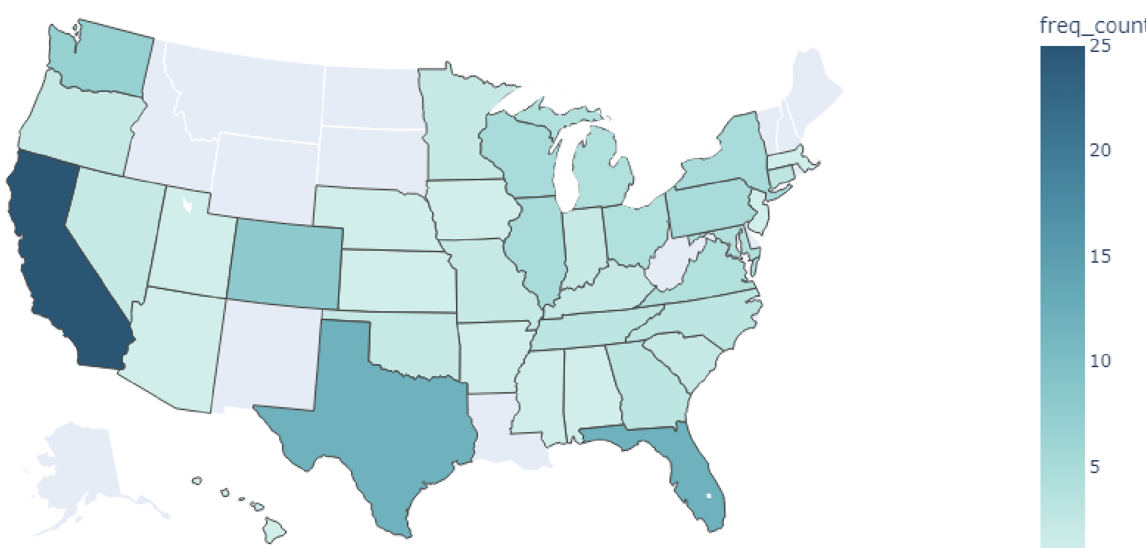


Hypothesis: 66.7% of weapons were obtained legally. Has access to weapons led to this state of things? Do we need stricter gun laws? We need more information and modeling to answer these questions.

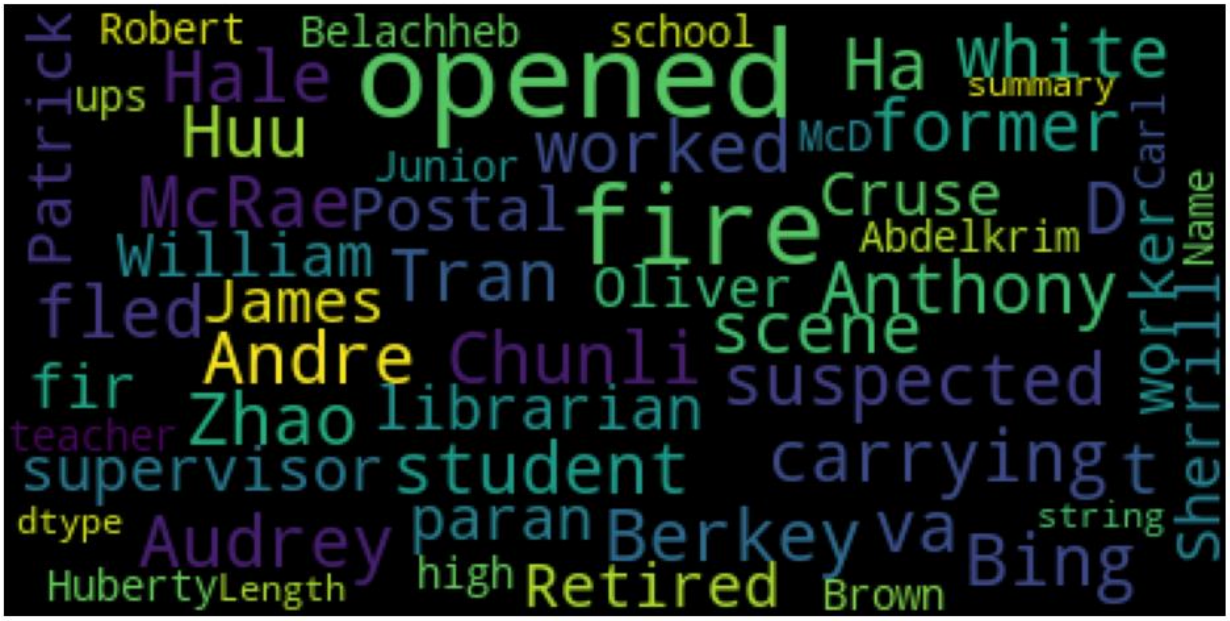
Module 2

Task 1: Exploratory Data Analysis

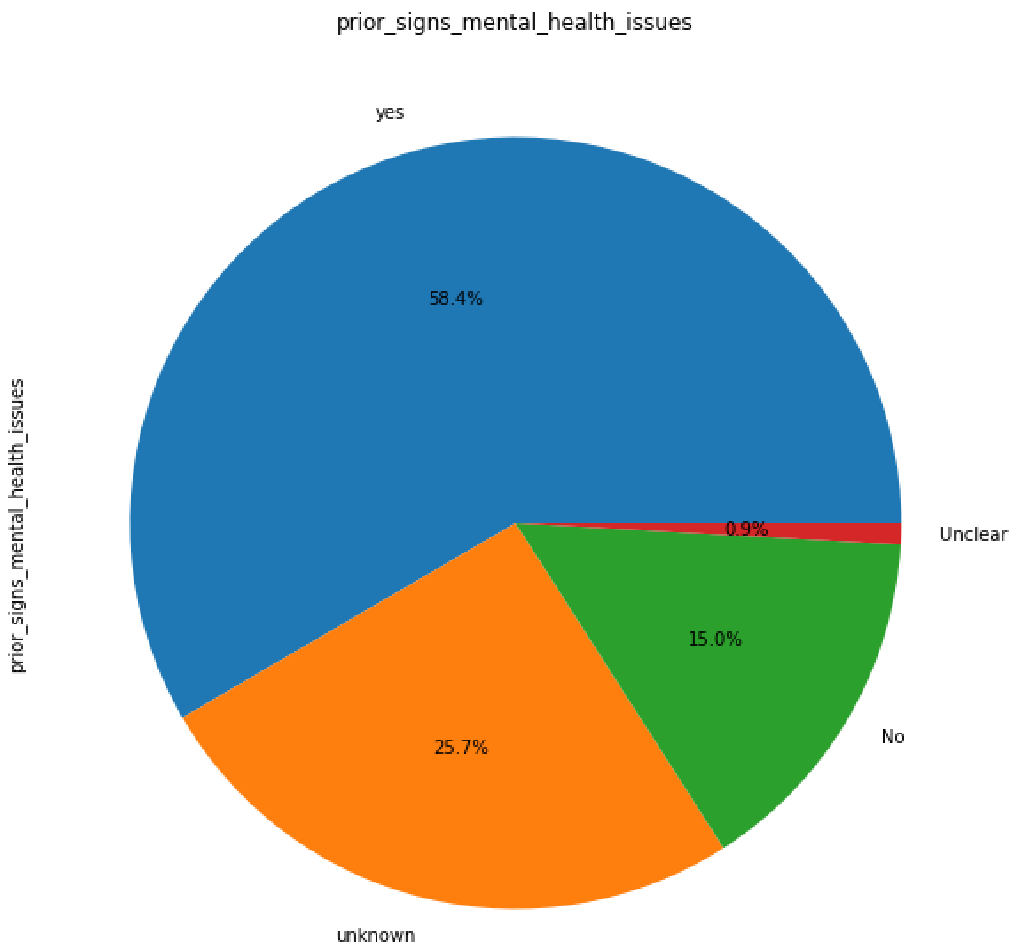
Step 1: To improve visualization I used Geojson to plot the number of cases on the map of the U.S. I had to link the US state and ID's dataset to mine to successfully get the visuals. This technique can easily help us refer that California, Florida, and Texas are the states most prone to Mass shootings. We can delve further into this by analyzing gun laws, population, and annual income in these states.

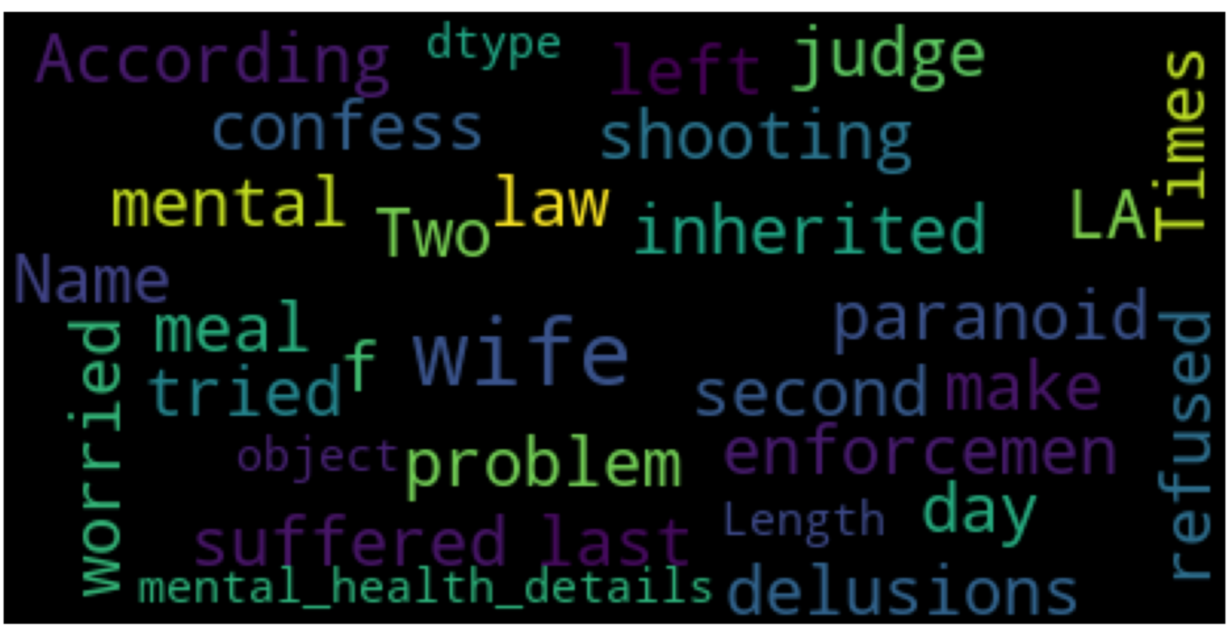


Step 2: To get an insight into the summary of the incidents without using NLP I tried to get an essence by using wordcloud. This gives us information about the locations like '**school'** and '**workplace'** being more prone. Maybe many of the shooters **fled** the **scene** after the incident.

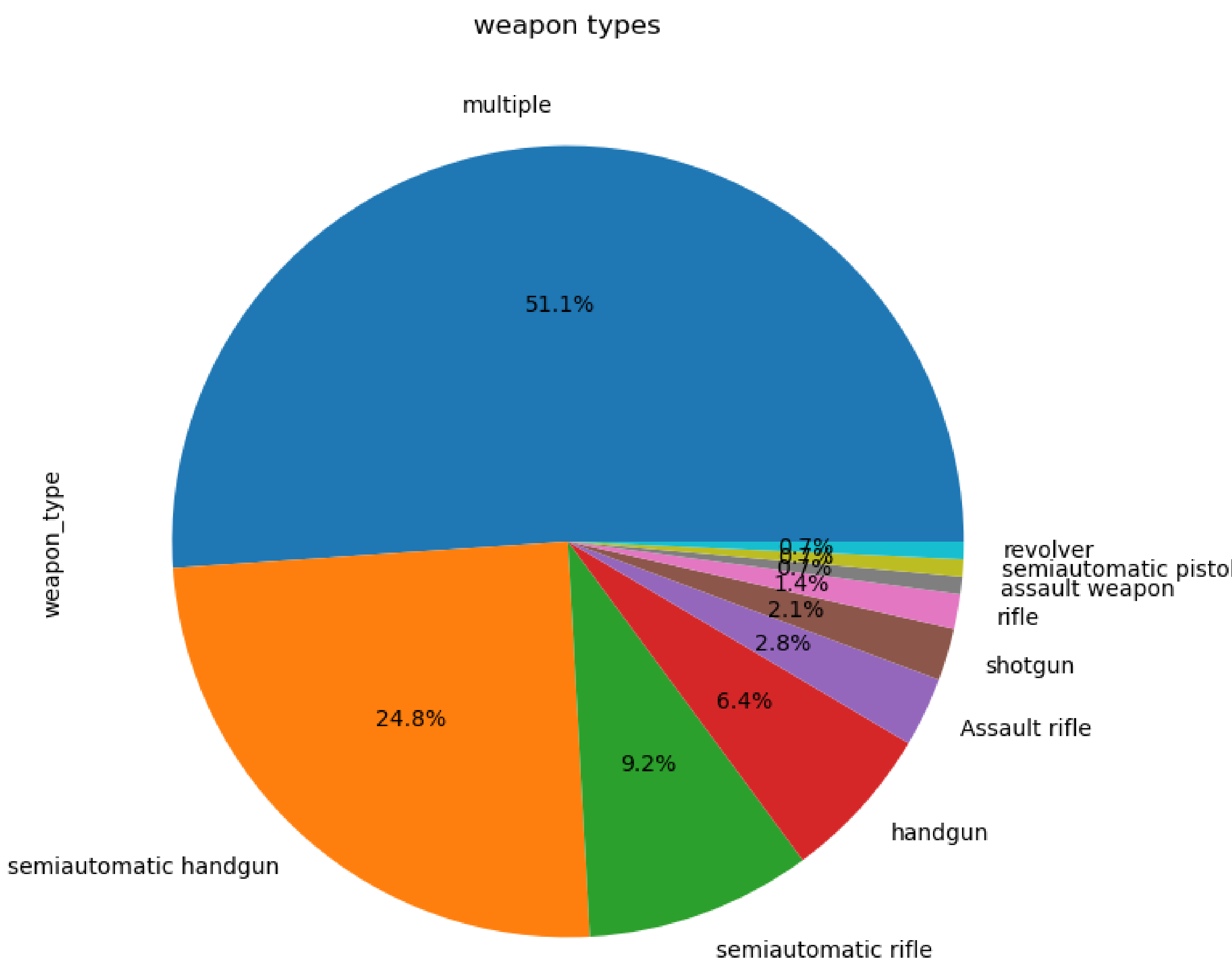


Step 3: To get an insight into the details of the mental health of the shooter(s) of the incidents without using NLP I tried to get an essence by using wordcloud. This gives us a deeper understanding without sentiment analysis. We can infer from the first chart that more than 50% of shooters suffer from prior mental health issues. Things like '**delusional'** and '**paranoid'** give us a sense of the suffering of the shooters and maybe some of them '**inherited'** these mental issues.


prior_signs_mental_health_issues
yes
58.4%
0.9%
Unclear
15.0%
No
25.7%
unknown
prior_signs_mental_health_issues


According
dtype
left
judge
Times
confess
shooting
mental
Two
law
inherited
LA
Name
paranoid
meal
wife
worried
tried
f
second
make
refused
object
problem
enforcemen
Length
day
suffered
last
mental_health_details
delusions

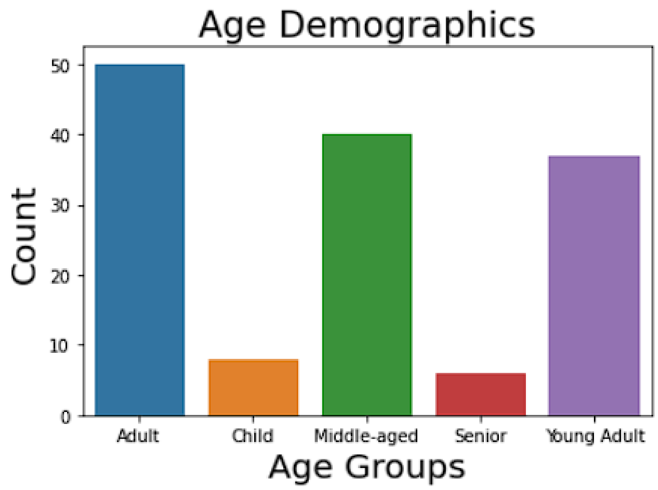
Step 4: In the previous phase we did not take much time looking at the weapons used and their details. So, I used a Pie chart to differentiate the types of weapons used.



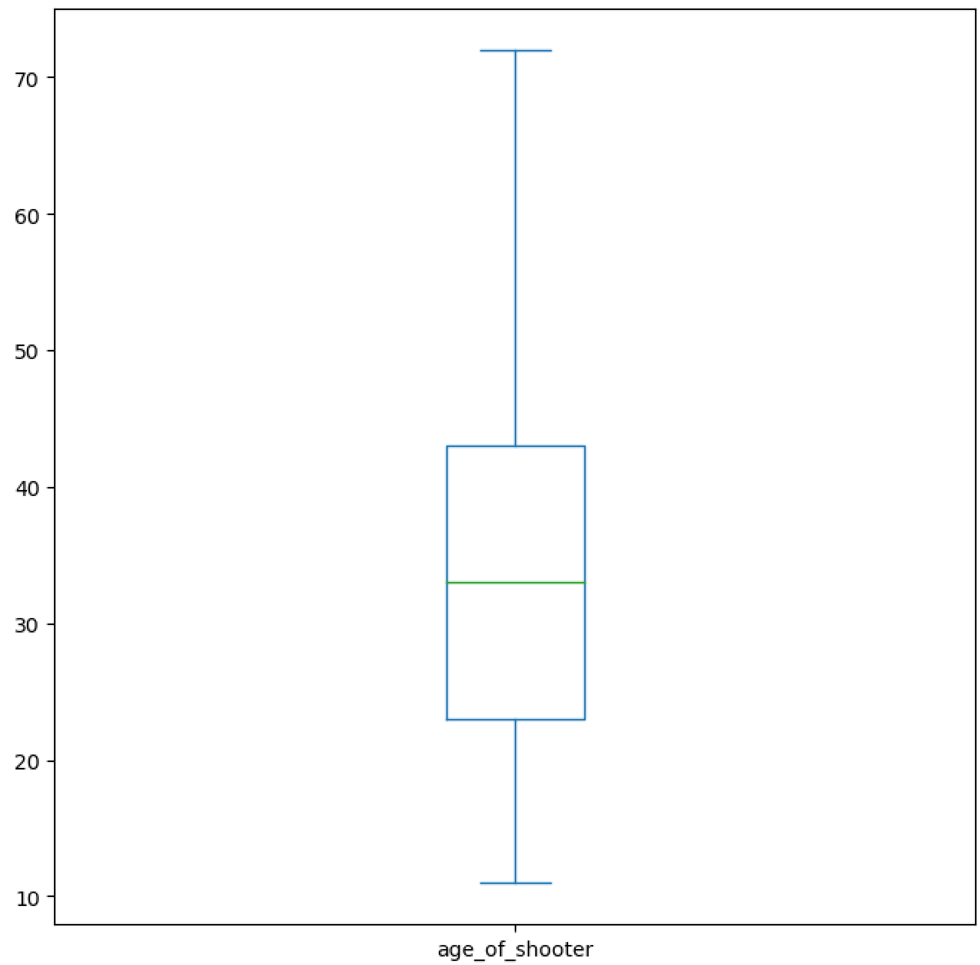
This information can help regulate the selling and buying of certain types of weapons.

Step 5: Previously we divided the ages into groups and the results were as follows:

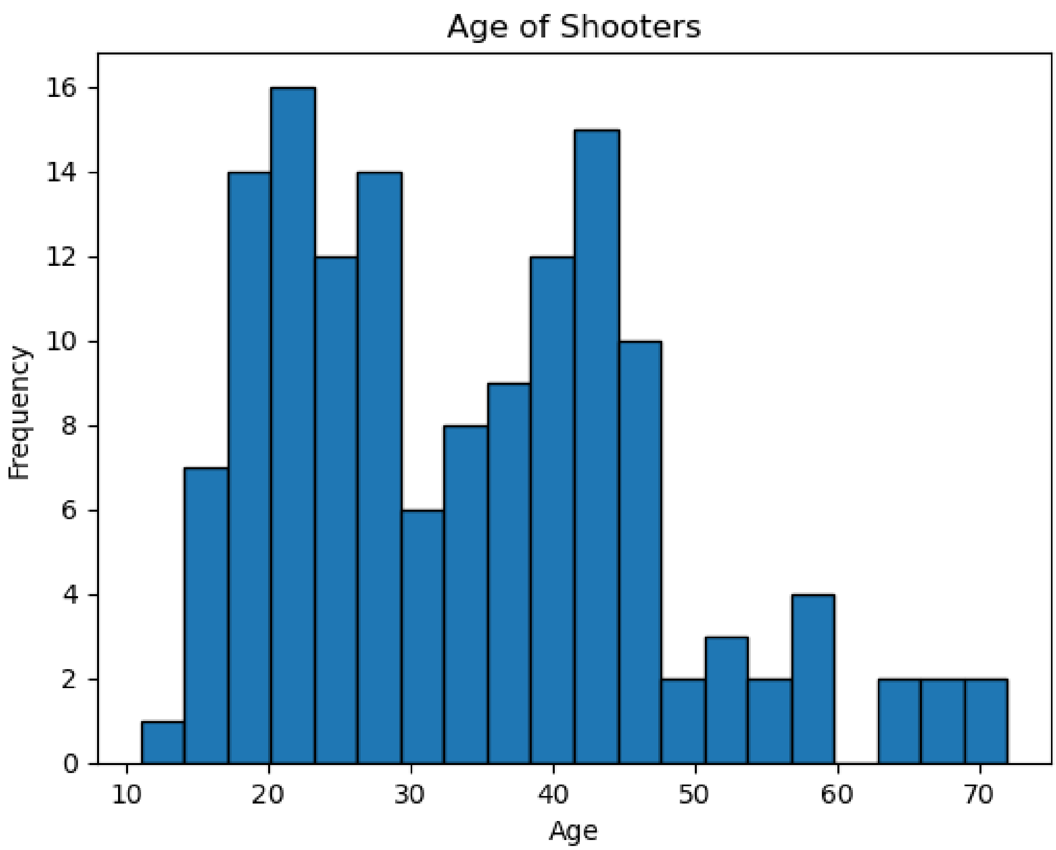
```
In [136]: Ages = df["Age Group"].sort_values(ascending=True)
          sns.countplot(x=Ages, data=df)
          plt.xlabel("Age Groups", fontsize=20)
          plt.ylabel("Count", fontsize=20)
          plt.title('Age Demographics', fontsize = 21)

Out[136]: Text(0.5, 1.0, 'Age Demographics')
```



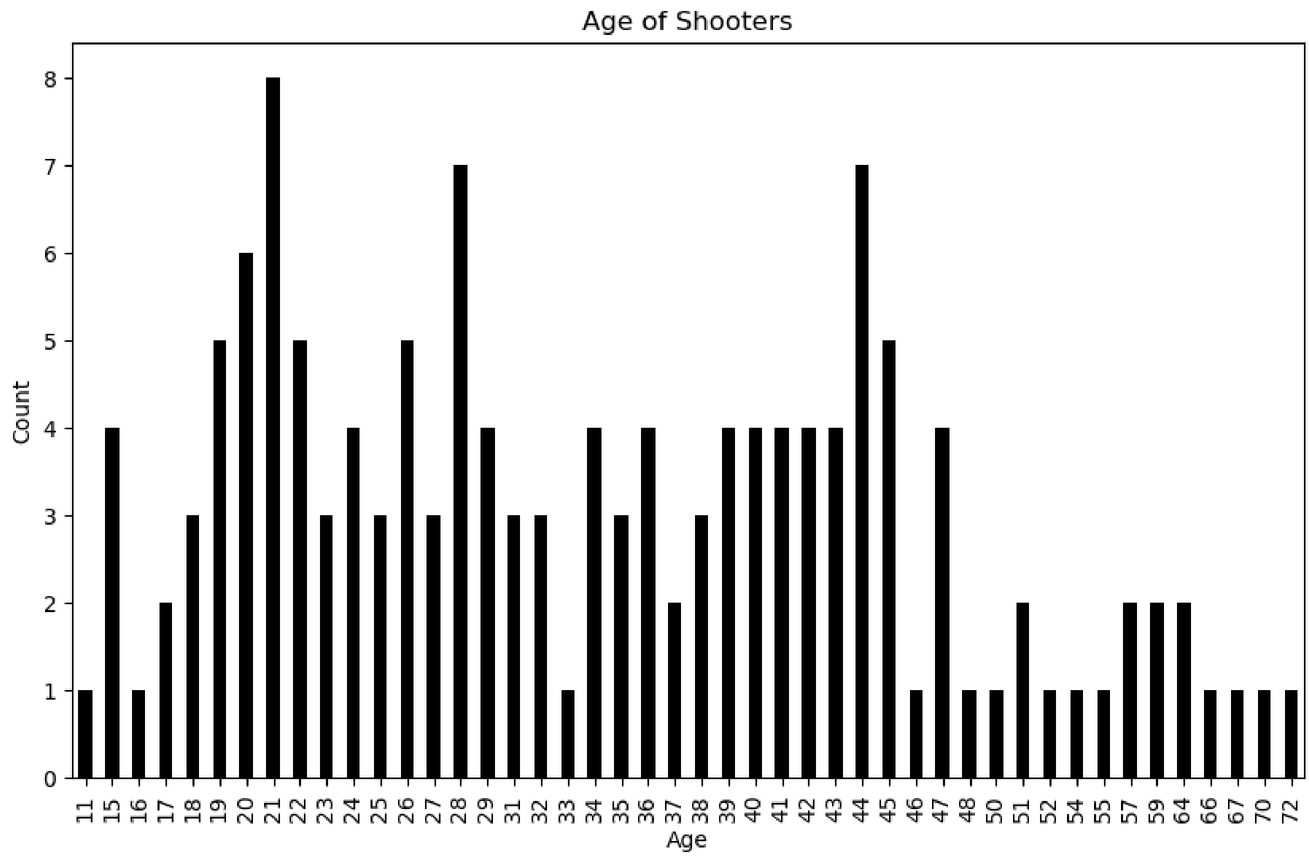
Going further into it, this time I tried to get more information about the individual ages using graphs. This box plot and histogram helps us get more information about ages. The oldest shooter is a little over 70 and the youngest shooter is as young as 11 years old.



70
60
50
40
30
20
10
age_of_shooter

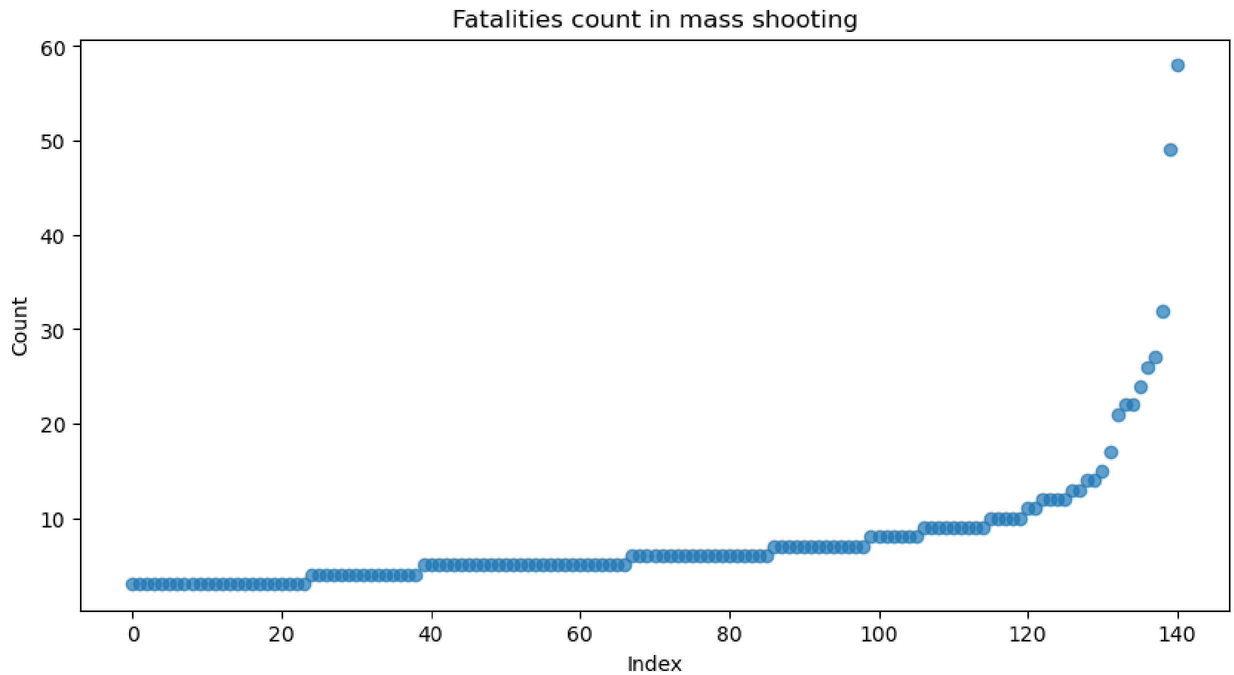


Step 6: Adding further to the previous step I plotted a bar chart to determine what ages are the most prone to getting involved in these crimes.

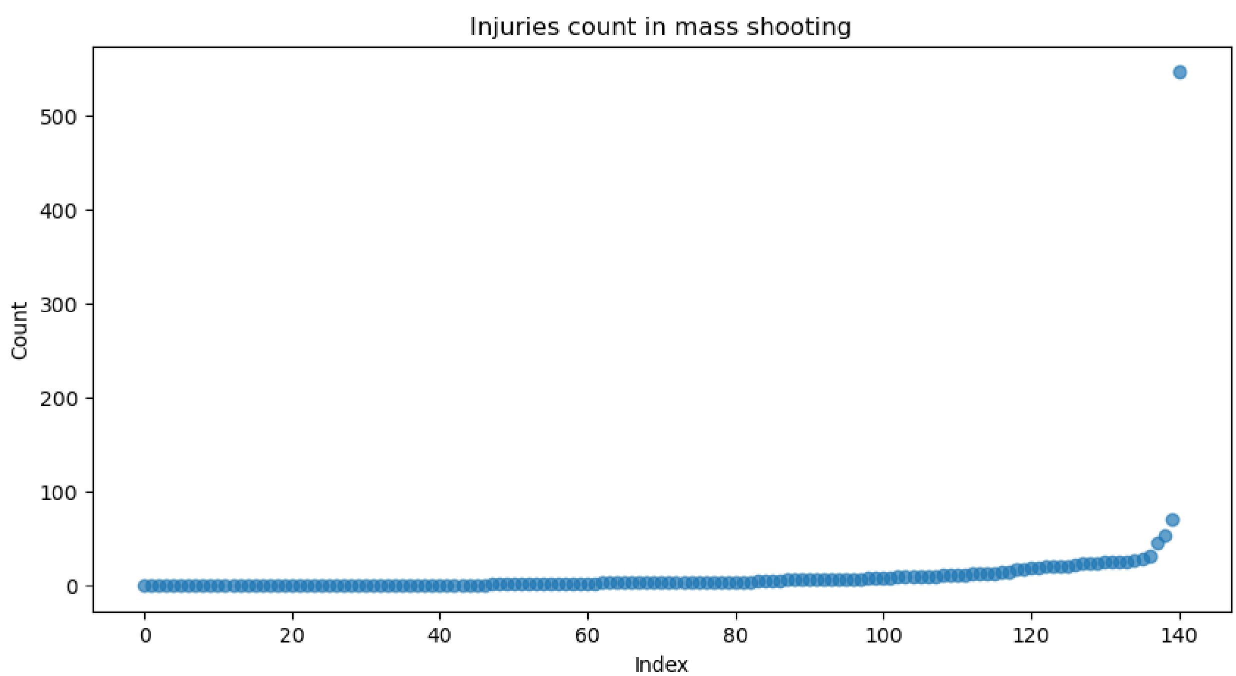


We see ages 19-22 have the most shooters with 21 being the highest. This information can help reform educational changes and have special programs for at-risk youths.

Step 7: To check outliers in data I tried plotting the fatalities count. This was done using a scatter plot to have easy visualization.

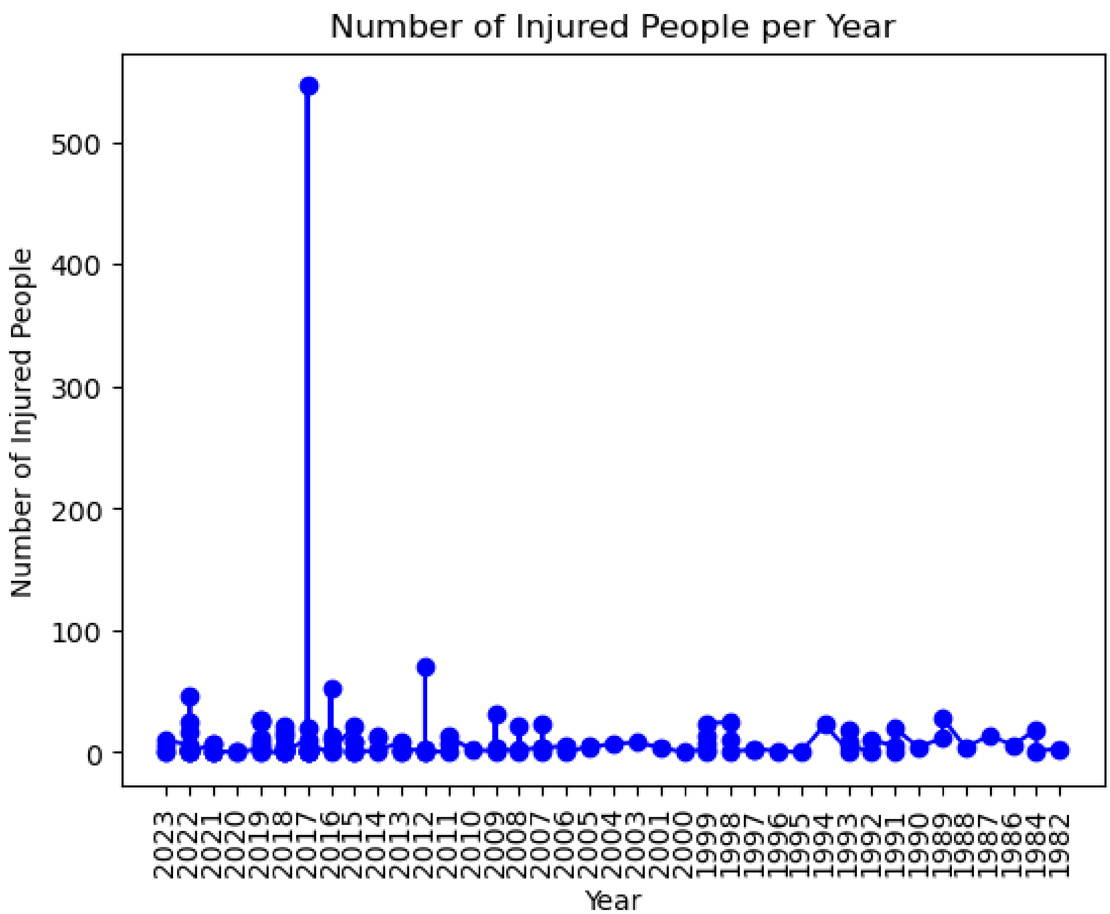


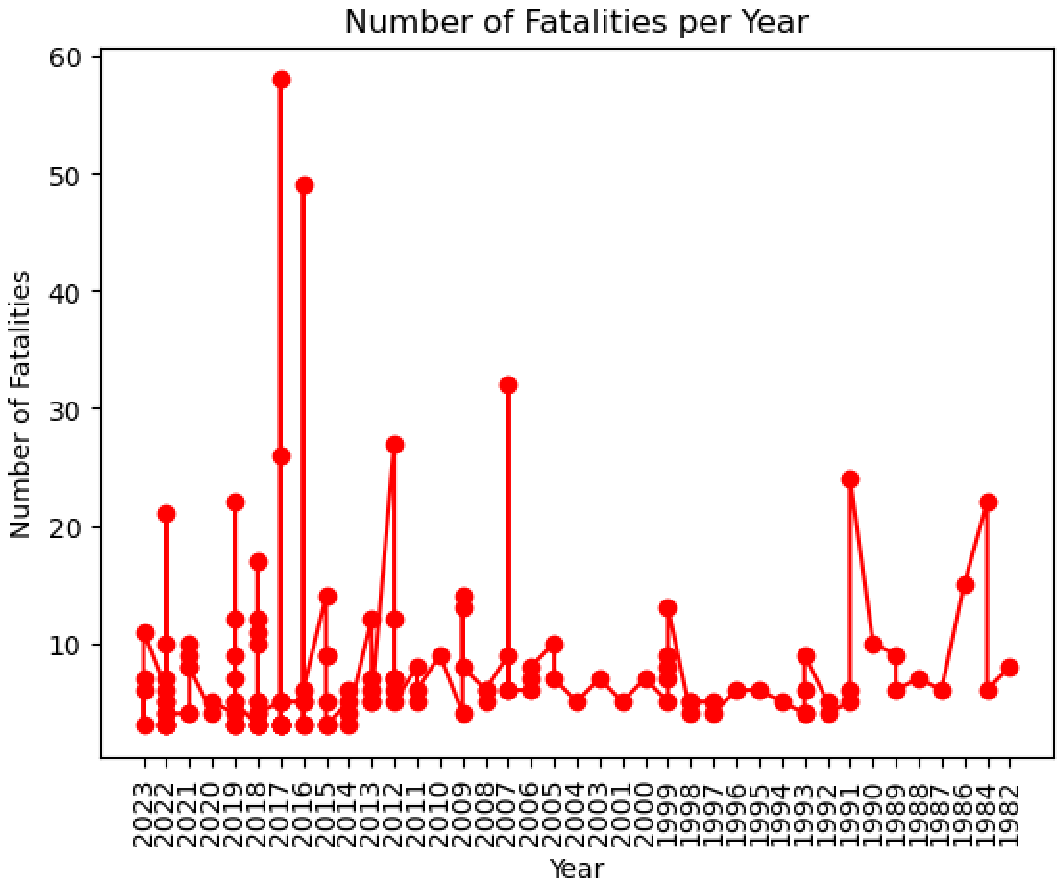
Step 8: To check outliers in data I tried plotting the injured count. This was done using a scatter plot to have easy visualization.



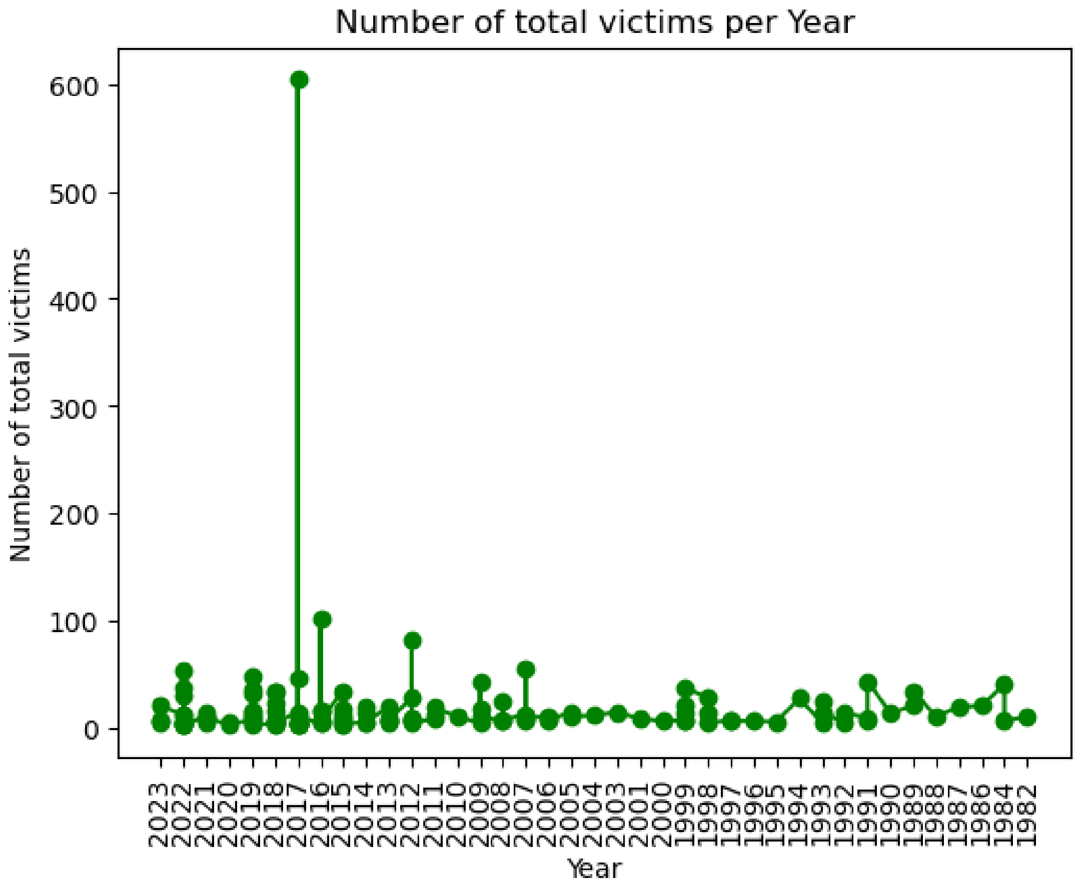
Checking outliers can help focus on incidents that result in many deaths/injuries to look closely into certain circumstances.

Step 9: Now to further analysis, we need to investigate what year these outlier incidents took place. Was there something specifically wrong about the years?





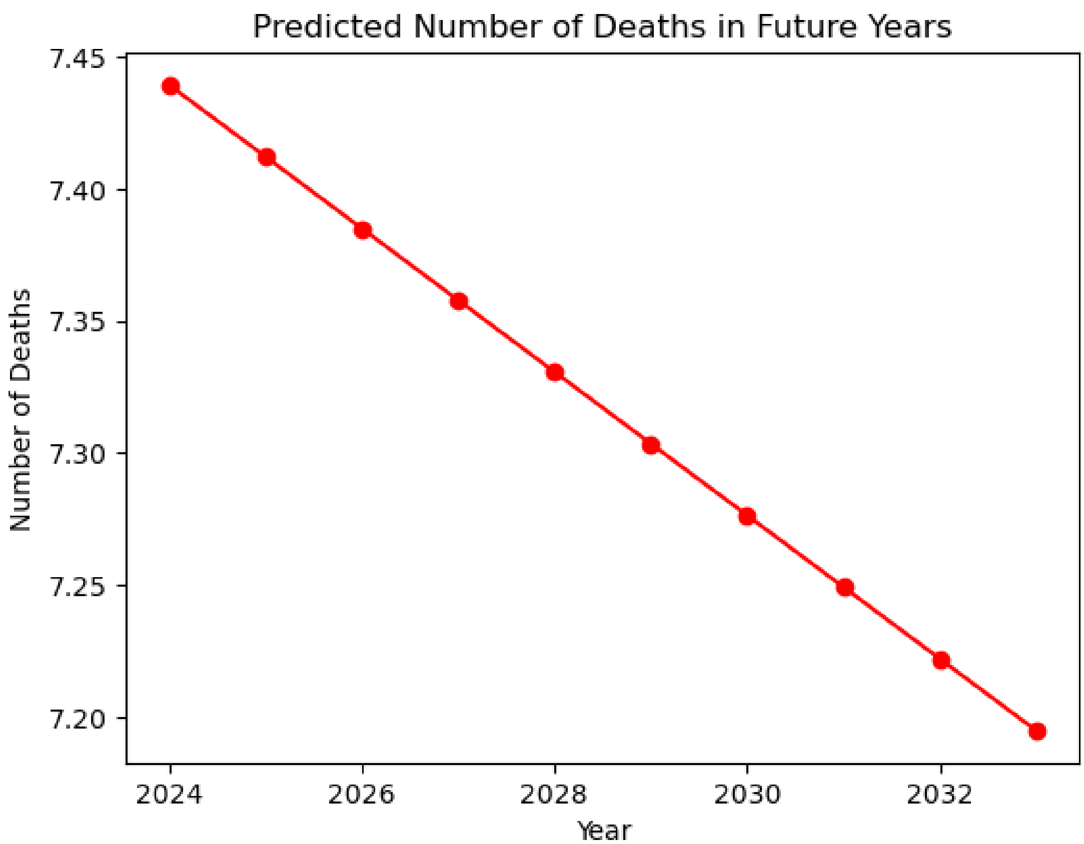
Step 10: To check the validity of the data I wanted to check if the total victim count aligns with the fatality and injured count.



<u>Task 2: Machine Learning Algorithms</u>

<u>Model 1: Linear Regression (Done in class)</u>

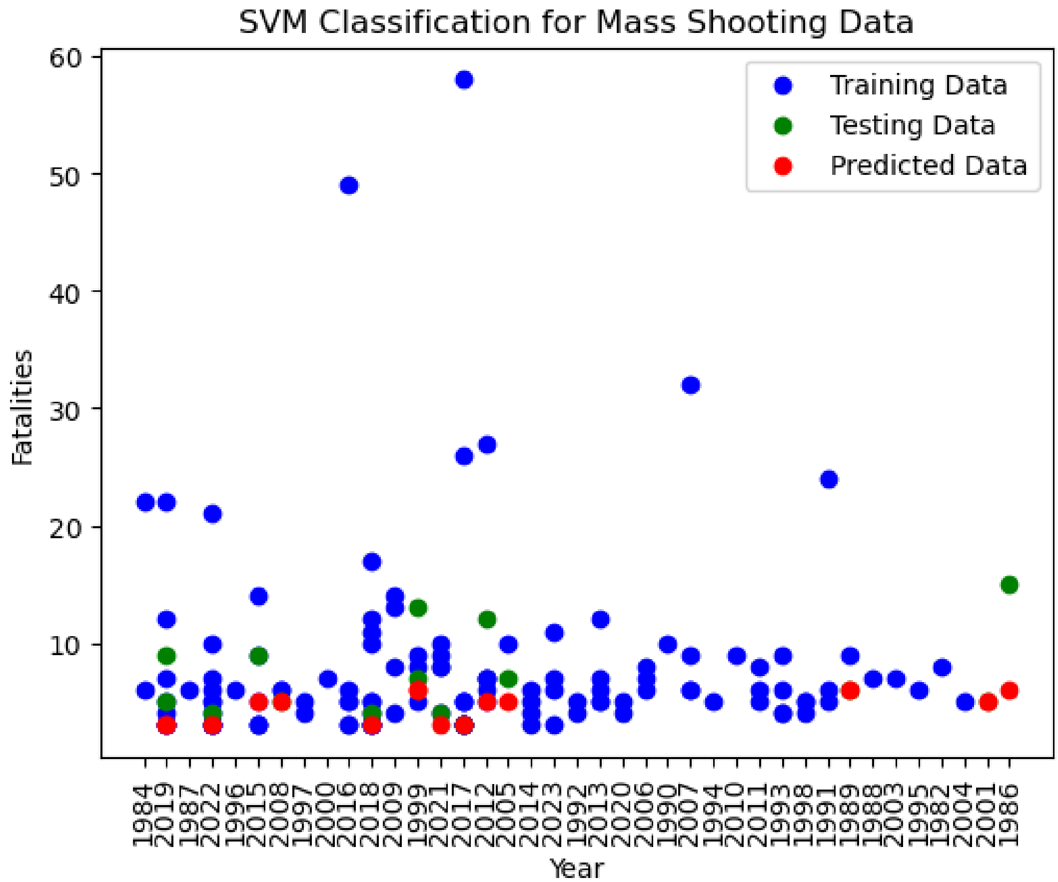
Using a simple linear regression, I wanted to try to predict the number of deaths that are likely to occur in the future years. The result is dependent on many subjective factors and may not be highly accurate. But a simple model based on current trends gave the following result.



The number of deaths is likely to be 7-8 people according to the model.

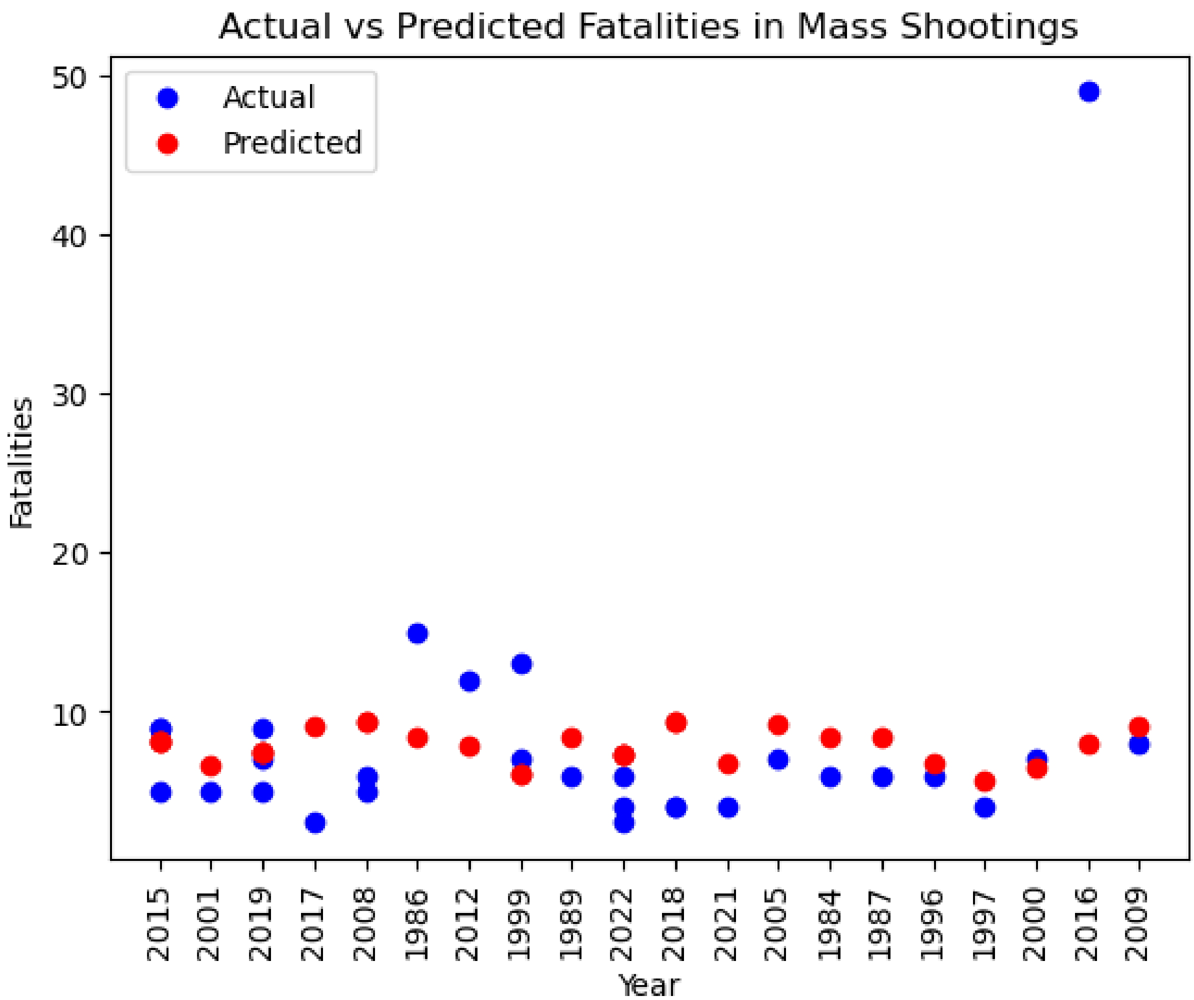
Model 2: Support Vector Machine (Not from class)

Support Vector Machine is a better model when we have a limited dataset and is likely to give more accurate results in a case where the data points are not that linear. SVM on fatalities gave the following results.



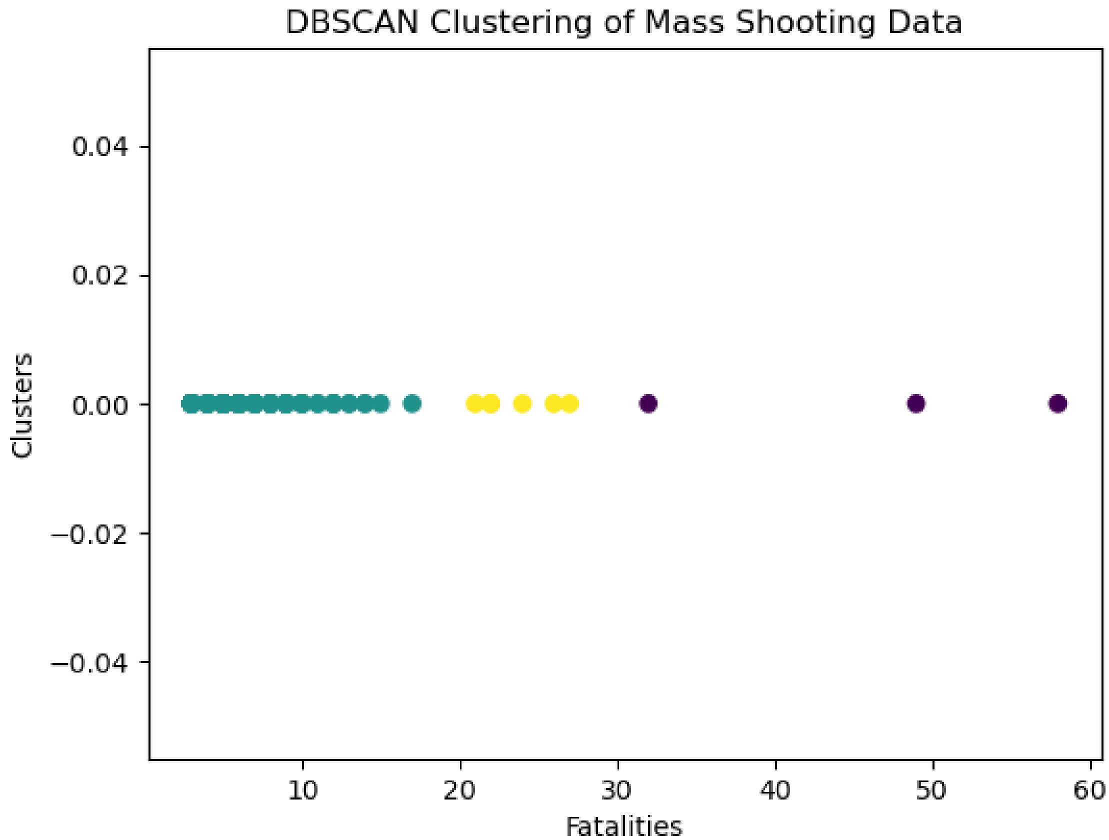
<u>Model 3: K-Nearest Neighbors (Done in Class)</u>

K-Nearest Neighbors was able to follow the trend quite accurately (better than the previous two models)



<u>Model 4: (DBSCAN) (Not from class)</u>

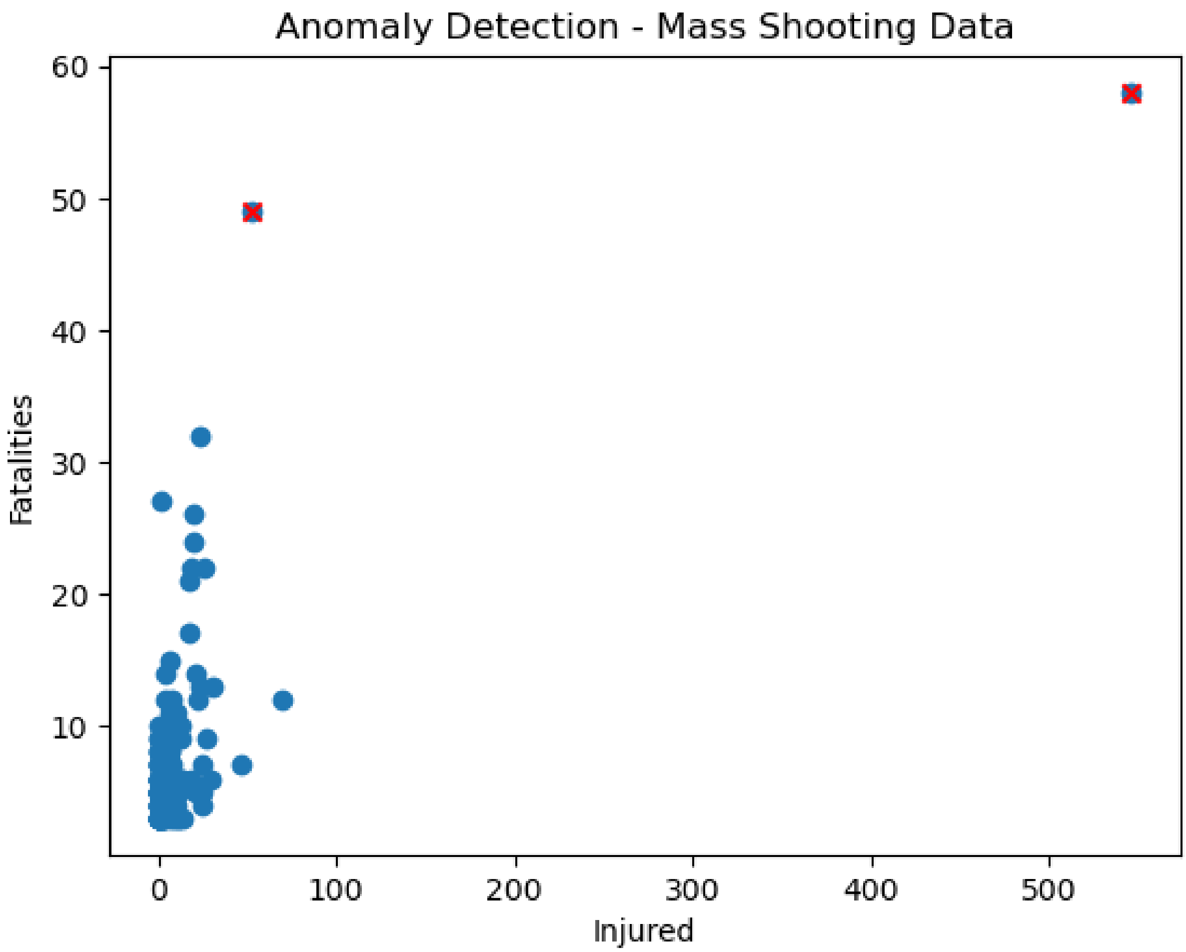
DBSCAN is a Density-Based clustering algorithm that is used when the data has noise(outliers). It is successfully able to combine densely grouped data points into one group. DBSCAN based on fatalities gave the following results:



<u>Model 5: Isolation Forest (Not from Class)</u>

Isolation Forest is an interesting anomaly detection algorithm that successfully detects anomalies from a dataset. This algorithm worked well on the dataset and gave the expected results. It detected anomalies on both injuries and fatalities.

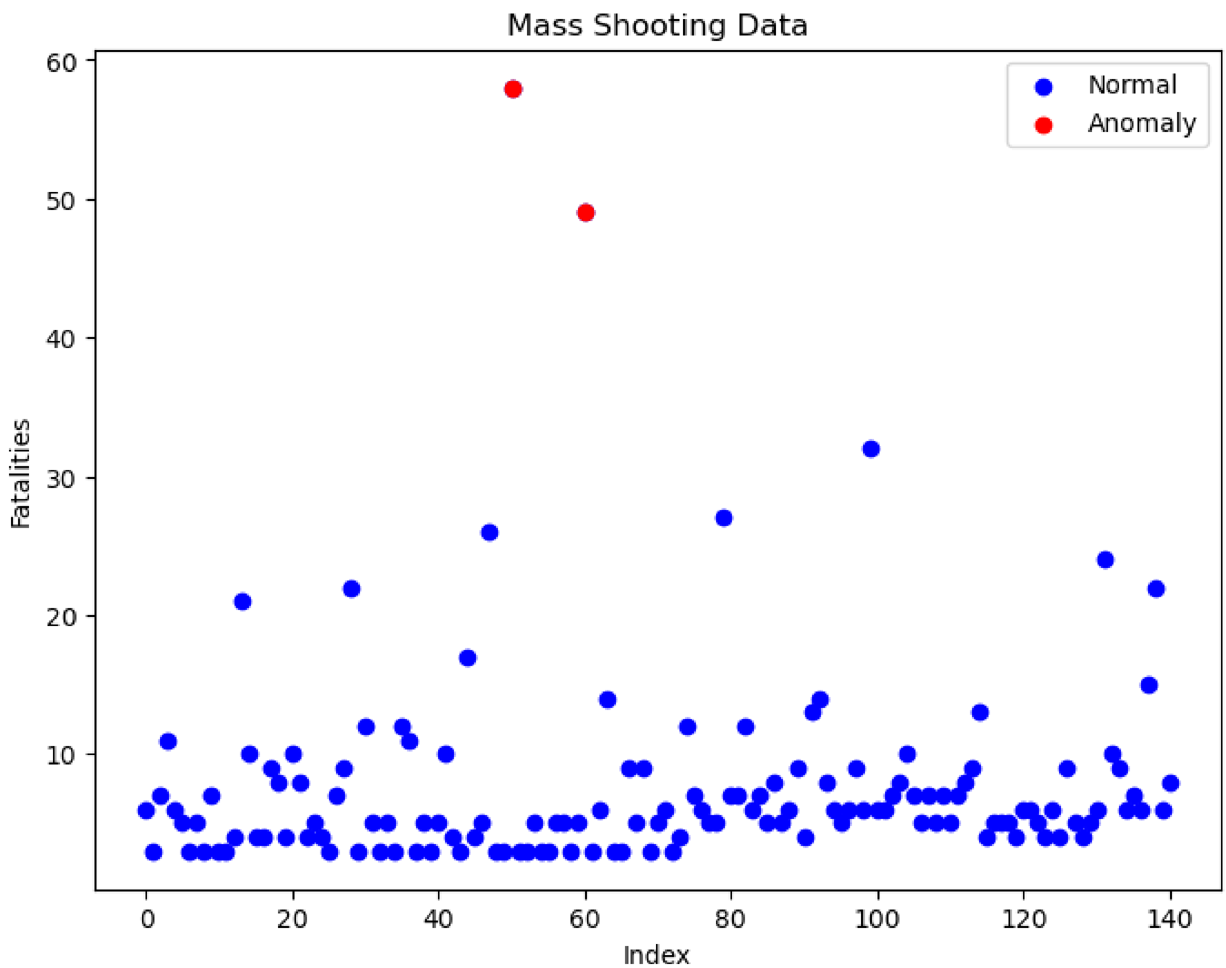
I also implemented isolation forest on these columns sepe



I also implemented IsolationForest on these columns separately. For fatalities, the results are as follows:

```
print(anomalies)
```

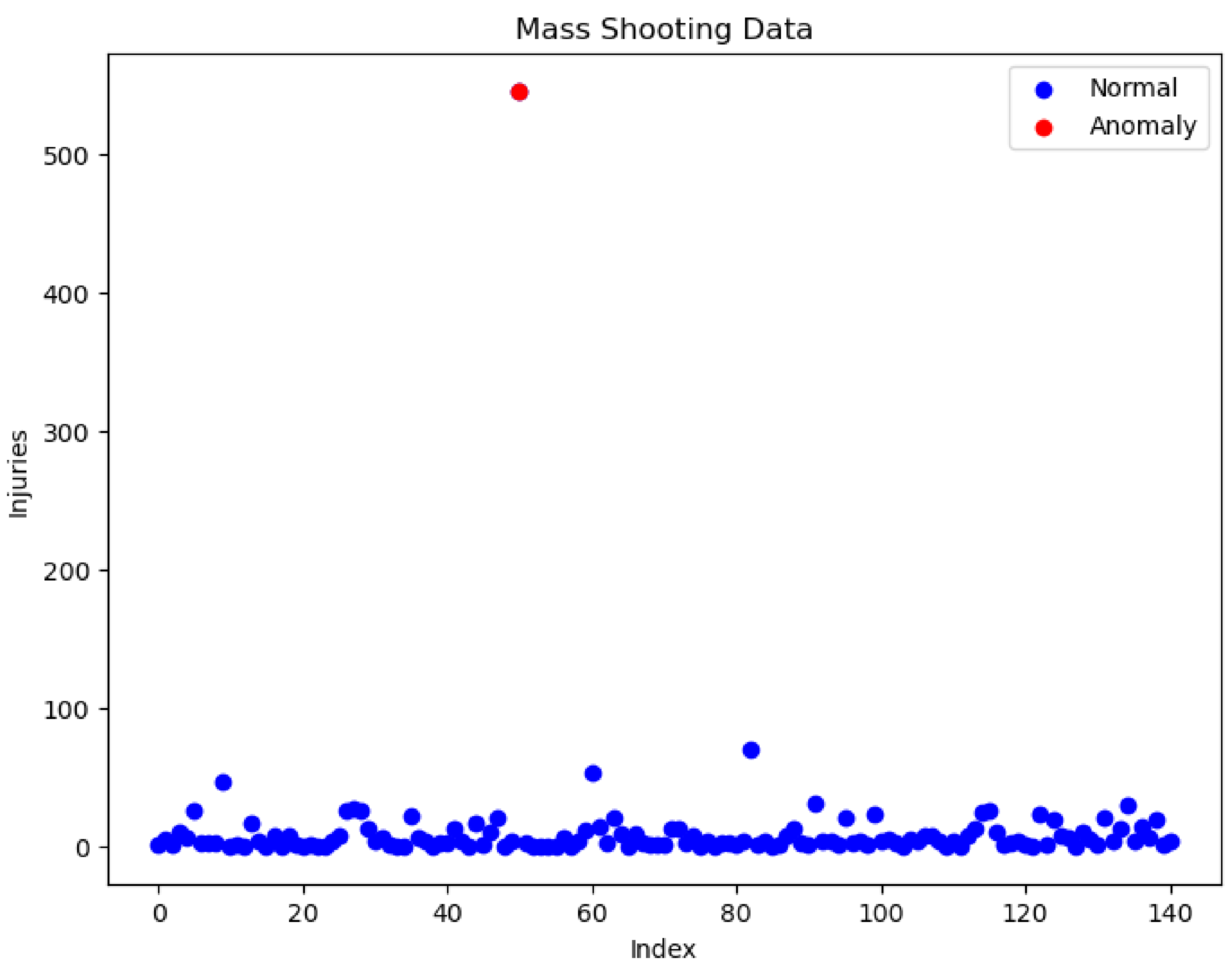
```
                         case          location       date  \
50    Las Vegas Strip massacre  Las Vegas, Nevada  10/1/2017   
60  Orlando nightclub massacre   Orlando, Florida  6/12/2016   

                                             summary  fatalities  injured  \
50  Stephen Craig Paddock, 64, fired a barrage of ...          58      546   
60  Omar Mateen, 29, attacked the Pulse nighclub i...          49       53   

    total_victims location.1  age_of_shooter prior_signs_mental_health_issues  \
50            604      other              64                          unknown   
60            102      other              29                          unknown   

    ... gender  type  year Month Day       City    State Age Group freq_count  \
50  ...   Male  Mass  2017    10   1  Las Vegas   Nevada    Senior          2   
60  ...   Male  Mass  2016     6  12    Orlando  Florida     Adult         12   

   anomaly_score  
50     -0.062541  
60     -0.022045  

[2 rows x 26 columns]
```



Similarly for Injured people as well the results are as follows:

```
scores = model.decision_function(X)
df["anomaly_score"] = scores
anomalies = df[df["anomaly_score"] < 0]
print(anomalies)
```

```
                         case           location       date  \
50  Las Vegas Strip massacre  Las Vegas, Nevada  10/1/2017

                                            summary  fatalities  injured  \
50  Stephen Craig Paddock, 64, fired a barrage of ...          58      546

    total_victims location.1  age_of_shooter prior_signs_mental_health_issues  \
50            604      other              64                          unknown

    ... gender  type  year Month Day       City   State Age Group freq_count  \
50  ...   Male  Mass  2017    10   1  Las Vegas  Nevada    Senior          2

   anomaly_score
50     -0.090464

[1 rows x 26 columns]
```



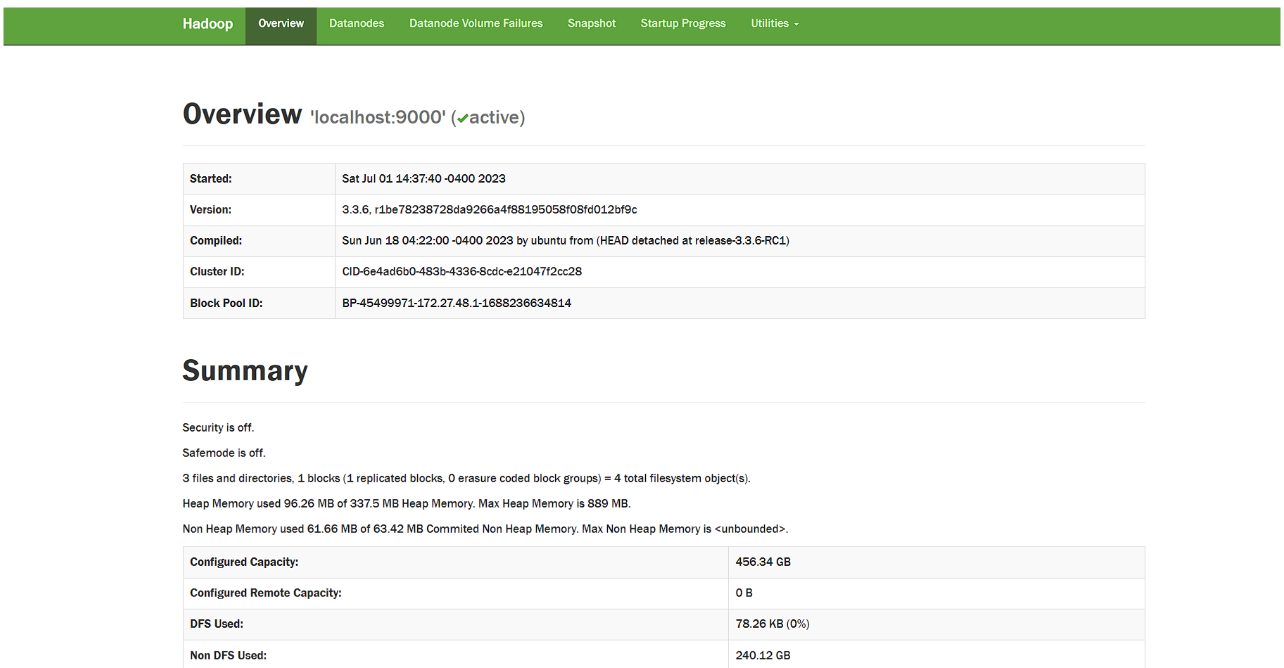
I used scatter plots for most of the data visualization as they seemed most suited for my dataset.

MODULE 3

TASK 1: LOADING DATA IN HADOOP

Hadoop setup – After successful installation of Hadoop I was able to create a local instance of Hadoop. Hadoop exhibits a master-slave structure, so I created a Namenode(Mater) and datanode(slave). I then loaded all the data into HDFS and Python and compared the time difference. Since the number of rows of my data does not exceed 2000, the difference in the times to load in Hadoop vs. Pandas isn't much.

NAMENODE INFORMATION:



Hadoop Overview Datanodes Datanode Volume Failures Snapshot Startup Progress Utilities

Overview 'localhost:9000' (active)

Started:	Sat Jul 01 14:37:40 -0400 2023
Version:	3.3.6, r1be78238728da9266a4f88195058f08fd012bf9c
Compiled:	Sun Jun 18 04:22:00 -0400 2023 by ubuntu from (HEAD detached at release-3.3.6-RC1)
Cluster ID:	CID-6e4ad6b0-483b-4336-8cdc-e21047f2cc28
Block Pool ID:	BP-45499971-172.27.48.1-1688236634814

Summary

Security is off.

Safemode is off.

3 files and directories, 1 blocks (1 replicated blocks, 0 erasure coded block groups) = 4 total filesystem object(s).

Heap Memory used 96.26 MB of 337.5 MB Heap Memory. Max Heap Memory is 889 MB.

Non Heap Memory used 61.66 MB of 63.42 MB Commited Non Heap Memory. Max Non Heap Memory is <unbounded>.

Configured Capacity:	456.34 GB
Configured Remote Capacity:	0 B
DFS Used:	78.26 KB (0%)
Non DFS Used:	240.12 GB

Number of Under-Replicated Blocks	0
Number of Blocks Pending Deletion (including replicas)	0
Block Deletion Start Time	Sat Jul 01 14:37:40 -0400 2023
Last Checkpoint Time	Sat Jul 01 14:37:15 -0400 2023
Enabled Erasure Coding Policies	RS-6-3-1024k

NameNode Journal Status

Current transaction ID: 8

Journal Manager	State
FileJournalManager(root=C:\hadoop-3.3.6\data\namenode)	EditLogFileOutputStream(C:\hadoop-3.3.6\data\namenode\current\edits_inprogress_0000000000000000001)

NameNode Storage

Storage Directory	Type	State
C:\hadoop-3.3.6\data\namenode	IMAGE_AND_EDITS	Active

DFS Storage Types

Storage Type	Configured Capacity	Capacity Used	Capacity Remaining	Block Pool Used	Nodes In Service
DISK	456.34 GB	78.26 KB (0%)	216.22 GB (47.38%)	78.26 KB	1

DATANODE INFORMATION:

Hadoop Overview Utilities

DataNode on Adikavya-Zephyrus.mshome.net:9866

Cluster ID:	CID-6e4ad6b0-483b-4336-8cdc-e21047f2cc28
Started:	Sat Jul 01 14:37:41 -0400 2023
Version:	3.3.6, r1be78238728da9266a4f88195058f08fd012bf9c

Block Pools

Namenode Address	Namenode HA State	Block Pool ID	Actor State	Last Heartbeat Sent	Last Heartbeat Response	Last Block Report	Last Block Report Size (Max Size)
localhost:9000	active	BP-45499971-172.27.48.1-1688236634814	RUNNING	1s	1s	4 hours	0 B (128 MB)

Volume Information

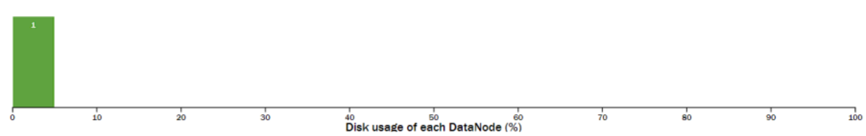
Directory	StorageType	Capacity Used	Capacity Left	Capacity Reserved	Reserved Space for Replicas	Blocks
C:\hadoop-3.3.6\data\datanode	DISK	78.26 KB	216.22 GB	0 B	0 B	1

Hadoop, 2023.

Datanode Information

In service Down Decommissioning Decommissioned Decommissioned & dead
Entering Maintenance In Maintenance In Maintenance & dead

Datanode usage histogram



In operation

DataNode State All Show 25 entries Search:

Node	Http Address	Last contact	Last Block Report	Used	Non DFS Used		Capacity	Blocks	Block pool used	Version
/default-rack/Adikavya-Zephyrus.mshome.net:9866 (127.0.0.1:9866)	http://Adikavya-Zephyrus.mshome.net:9864	0s	265m	78.26 KB	240.11 GB	456.34 GB		1	78.26 KB (0%)	3.3.6

Showing 1 to 1 of 1 entries Previous 1 Next

Data loading in python:

```
In [53]: import pandas as pd
         import time
         from pathlib import Path
         start_time = time.time()

         my_csv = Path("C:/Users/adika/OneDrive/Desktop/ub/summer23/DIC_587/project/shooting-1982-2023.csv")
         df = pd.read_csv(my_csv.resolve(), sep=',')

         end_time = time.time()
         elapsed_time = end_time - start_time

         print("Time taken to load the CSV file:", elapsed_time, "seconds")

         Time taken to load the CSV file: 0.029169797897338867 seconds
```

<u>Loading DATA in a local instance of HADOOP:</u>

```
C:\hadoop-3.3.6\sbin>hadoop fs -put C:\hadoop-3.3.6\shooting-1982-2023.csv

C:\hadoop-3.3.6\sbin>hadoop fs -put  C:\hadoop-3.3.6\shooting-1982-2023.csv /input
put: `C:/hadoop-3.3.6/shooting-1982-2023.csvv': No such file or directory

C:\hadoop-3.3.6\sbin>hadoop fs -put C:\hadoop-3.3.6\shooting-1982-2023.csv /input

C:\hadoop-3.3.6\sbin>hadoop fs -ls /input
Found 1 items
-rw-r--r--   1 adika supergroup      79192 2023-07-01 19:19 /input/shooting-1982-2023.csv

C:\hadoop-3.3.6\sbin>
```

```
C:\hadoop-3.3.6\sbin>hadoop fs -cat /input/shooting-1982-2023.csv
n╗┐case,location,date,summary,fatalities,injured,total_victims,location,age_of_shooter,prior_signs_mental_health_issues,mental_health_details,weapons_obtained_legally,where
_obtained,weapon_type,weapon_details,race,gender,type,year
Nashville religious school shooting,"Nashville, Tennessee",3/27/2023,"Audrey Hale, 28, who was a former student at the private Covenant School, killed three adults and thre
e 9-year-old children, before dying in a shootout with police.",6,1,6,School,28,,-,unknown,-,multiple,-,,"F (identified as transgender, per police)",Mass,2023
Michigan State University shooting,"East Lansing, Michigan",2/13/2023,"Anthony D. McRae, 43, opened fire at Berkey Hall and the MSU union, according to local police. Follow
ing an intense manhunt in the area, he was found dead from a self-inflicted gunshot wound, police said.",3,5,8,School,43,,-,yes,-,multiple,-,Black,M,Mass,2023
Half Moon Bay spree shooting,"Half Moon Bay, California",1/23/2023,"Chunli Zhao, 67, suspected of carrying out the attacks at a mushroom farm and near a trucking facility,
was apprehended by police. Zhao reportedly worked at the mushroom farm.",7,1,8,work,67,,-,unknown,-,semiautomatic handgun,-,Asian,M,Spree,2023
LA dance studio mass shooting,"Monterey Park, California",1/21/2023,"Huu Can Tran, 72,┬áfled the scene in a white van and later shot himself to death as police closed in.",
11,10,21,Other,72,yes,"According to the LA Times, ""Two law enforcement sources said the suspect recently showed up to the Hemet police station saying his family was trying
 to poison him.""",unknown,-,semiautomatic assault weapon (Details pending),-,Asian,M,Mass,2023
Virginia Walmart shooting,"Chesapeake, Virginia",11/22/2022,"Andre Bing, 31, who worked as a supervisor at the store, opened fire on co-workers and then fatally shot himsel
f, according to local authorities.",6,6,12,work,31,,-,unknown,-,semiautomatic handgun,-,Black,M,Mass,2022
LGBTQ club shooting,"Colorado Springs, Colorado",11/19/2022,"Anderson L. Aldrich, 22, wore body armor and opened fire upon entering the club as a dance party was underway;
he was subdued by unarmed patrons who tackled him amid the carnage and held him down until police arrived.",5,25,30,Other,22,yes,Aldrich reportedly had a history of menacin
g behavior and violent threats.,unknown,-,multiple,-,White,M,Mass,2022
University of Virginia shooting,"Charlottesville, Virginia",11/13/2022,"Christopher Darnell Jones Jr., 22, allegedly opened fire after a charter bus returned to campus from
 a university field trip, killing three members of the UVA football team and injuring two other people. Jones Jr. reportedly was on the radar of the university's threat ass
essment team regarding talk of owning a gun and a prior 2021 incident involving a concealed weapon.",3,2,5,School,22,,-,yes,"Dance's Sporting Goods; Colonial Heights, VA",s
emiautomatic pistol,"Glock 45 9mm; Ruger AR-556 rifle (in his dorm room, with other weapons, gear, and ammo)",Black,M,Mass,2022
Raleigh spree shooting,"Hedingham, North Carolina",10/13/2022,"Austin Thompson, 15, went on a rampage in the Hedingham neighborhood, where he lived; one of the fatalities i
ncluded his 16-year-old brother, James. Thompson was critically wounded and apprehended by police after a long standoff, and was admitted to an area hospital in critical co
ndition.",5,2,7,Other,15,,-,unknown,-,multiple,,White,M,Spree,2022
```

Browse Directory

/input Go!

Show 25 entries Search:

	Permission	Owner	Group	Size	Last Modified	Replication	Block Size	Name
☐	-rw-r--r--	adika	supergroup	77.34 KB	Jul 01 19:19	1	128 MB	shooting-1982-2023.csv

Showing 1 to 1 of 1 entries

Previous 1 Next

Hadoop, 2023.

The time taken to load data in **JUPYTER NOTEBOOK** was **0.029 seconds** while the time taken to load the dataset in **Hadoop** was **0.008 seconds**.

TASK – II: Wordcount on data

To get more insights on the data I decided to perform WORDCOUNT on the column **'Summary'** (feature selection) of my dataset. This column contains the incident summary and analyzing the most used words can help give us more information about our data.
On performing word count on data without using MapReduce the following results were obtained.

```python
df = df.dropna(subset=['summary'])
df['summary'] = df['summary'].str.lower()
df['summary'] = df['summary'].str.split()

word_count = {}

for row in df['summary']:
    for word in row:
        word_count[word] = word_count.get(word, 0) + 1

word_count_df = pd.DataFrame.from_dict(word_count, orient='index', columns=['Frequency'])
word_count_df.index.name = 'Word'
word_count_df = word_count_df.sort_values(by='Frequency', ascending=False)
print(word_count_df)

end_time = time.time()
elapsed_time = end_time - start_time
print("Time taken to perform word count", elapsed_time, "seconds")
```

```
javier              1
shots.)             1
(no                 1
floor;              1
moseley,            1
snochia             1
elections.          1
2018                1
content,            1
ahead               1
republican          1
trump               1
hyped               1
"invaders"          1
caravan             1
migrant             1
references          1
welding             1
Time taken to perform word count 0.027048110961914062 seconds
```

Word	Frequency
'a',	283
'the',	212
'and',	191
'in',	118
'he',	110
'at',	98
'to',	97
'was',	94
'of',	74
'fire',	71
'his',	69
'opened',	69
'before',	64
'shot',	64
'by',	54
'police',	53
'with',	51
'on',	49
'killed',	44
'an',	42
'after',	40
'had',	39
'then',	31
'who',	31
'three',	28
'as',	28
'killing',	27
'two',	26
'later',	23
'from',	23
'suicide.']	23
'for',	22
'fatally',	21
'committing',	21
'people',	20

On performing the same on MapReduce, the following error was obtained:

```
C:\hadoop-3.3.6>start-dfs

C:\hadoop-3.3.6>start-yarn
starting yarn daemons

C:\hadoop-3.3.6>hdfs dfs -put c:/hadoop-3.3.6/myfile.txt /input

C:\hadoop-3.3.6>bin\yarn jar share/hadoop/mapreduce/hadoop-mapreduce-examples-3.3.6.jar wordcount hdfs://localhost:9870/user/adika/input output
2023-07-04 01:44:47,785 INFO client.DefaultNoHARMFailoverProxyProvider: Connecting to ResourceManager at /0.0.0.0:8032
2023-07-04 01:44:48,276 INFO mapreduce.JobResourceUploader: Disabling Erasure Coding for path: /tmp/hadoop-yarn/staging/adika/.staging/job_1688449435131_0001
2023-07-04 01:44:48,494 INFO mapreduce.JobSubmitter: Cleaning up the staging area /tmp/hadoop-yarn/staging/adika/.staging/job_1688449435131_0001
org.apache.hadoop.ipc.RpcException: RPC response exceeds maximum data length
        at org.apache.hadoop.ipc.Client$IpcStreams.readResponse(Client.java:1920)
        at org.apache.hadoop.ipc.Client$Connection.receiveRpcResponse(Client.java:1187)
        at org.apache.hadoop.ipc.Client$Connection.run(Client.java:1078)

C:\hadoop-3.3.6>stop-all
This script is Deprecated. Instead use stop-dfs.cmd and stop-yarn.cmd
SUCCESS: Sent termination signal to the process with PID 38184.
SUCCESS: Sent termination signal to the process with PID 18764.
stopping yarn daemons
SUCCESS: Sent termination signal to the process with PID 11368.
SUCCESS: Sent termination signal to the process with PID 34088.

INFO: No tasks running with the specified criteria.

C:\hadoop-3.3.6>start-dfs

C:\hadoop-3.3.6>start-yarn
starting yarn daemons

C:\hadoop-3.3.6>bin\yarn jar share/hadoop/mapreduce/hadoop-mapreduce-examples-3.3.6.jar wordcount hdfs://localhost:9870/user/adika/input output
```

Since the code wasn't working on Hadoop a time comparison couldn't be performed but again due to the dataset only having a few rows I assume the time would've been similar in both cases. The time taken to perform **wordcount without MapReduce is 0.027 seconds**.

TASK – III Working of MapReduce for Wordcount on 'summary' data:

Map Stage

1. Input: The .csv file containing the incident summaries.
2. Mapper: Each line of the .csv file is processed by the mapper, which extracts the "Summary" column value.
3. Tokenization: The mapper tokenizes the summary into individual words, discarding punctuation and converting all words to lowercase.
4. Key-Value Pair Emission: The mapper emits key-value pairs, where the key is each word from the summary, and the value is the number '1'.

Reduce Stage

1. Shuffle and Sort: The framework groups together the key-value pairs based on the key and sorts them by the key.
2. Reducer: Each unique word is received by the reducer.
3. Count Aggregation: The reducer counts the occurrences of each word by summing the corresponding values (1s) received for each key.
4. Output: The reducer emits the word and its count as the final output.

In-depth working:

Map Stage:

Input: The .csv file with the "Summary" column.

Mapper: Each mapper task processes one line at a time, obtaining the "Summary" column value.

For example, the first mapper task processes the first line:
Input: "A mass shooting occurred in Townsville. Several people in there were injured."

Tokenization: The mapper tokenizes the summary into individual words and converts them to lowercase, discarding punctuation.

Output Key-Value Pairs:
Key: "a", Value: 1
Key: "mass", Value: 1
Key: "shooting", Value: 1
Key: "occurred", Value: 1
Key: "in", Value: 1
Key: "townsville", Value: 1
Key: "several", Value: 1
Key: "people", Value: 1

Key: "were", Value: 1
Key: "injured", Value: 1
Key: “there”, Value:1

Shuffle and Sort: The framework groups and sorts the key-value pairs based on the key.

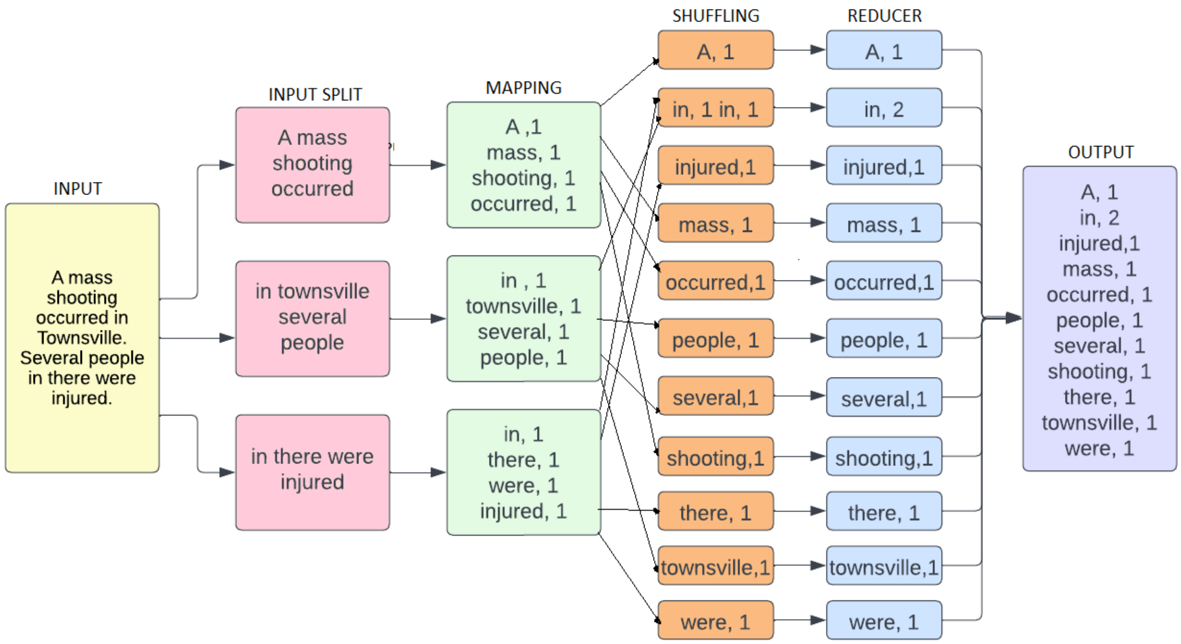
Sorted Key-Value Pairs:
Key: "a", Value: [1]
Key: "in", Value: [1, 1]
Key: "mass", Value: [1]
Key: "occurred", Value: [1]
Key: "people", Value: [1]
Key: "several", Value: [1]
Key: "shooting", Value: [1]
Key: "townsville", Value: [1]
Key: "were", Value: [1]
Key: "injured", Value: [1]
Key: “there”, Value:[1]

Reducer: Each reducer task receives the sorted key-value pairs for a unique word.
For e.g., the reducer for the word "in" receives [1, 1].

Count Aggregation: The reducer sums the values to calculate the count for each word.
Output Key-Value Pair (for the word "in"):
Key: "in", Value: 2

The final output for the word count of the "Summary" column would be:
"a": 1
"mass": 1
"shooting": 1
"occurred": 1
"in": 2
"townsville": 1
"several": 1
"people": 1
"were": 1
"injured": 1
“there”:1

Here is a diagram to understand the architecture of MapReduce:



TASK – IV EXTRA CREDIT (SENTIMENT ANALYSIS):

I Performed sentiment analysis on the column "mental_health_details" of my dataset using word2vec and the results were as expected. All except one row was negative or neutral.

```
    words = nltk.word_tokenize(text)
    word_vectors = average_word_vectors(words, word2vec_model, word2vec_model.wv.key_to_index, 100)
    sentiment_score = sid.polarity_scores(" ".join(words))['compound']
    if sentiment_score >= 0.05:
        return 'Positive'
    elif sentiment_score <= -0.05:
        return 'Negative'
    else:
        return 'Neutral'

df['sentiment'] = df['mental_health_details'].apply(sentiment_analysis)
```

```
In [58]: print(df[['case','sentiment']])
```

```
                                         case sentiment
0           Nashville religious school shooting   Neutral
1          Michigan State University shooting   Neutral
2                 Half Moon Bay spree shooting   Neutral
3                LA dance studio mass shooting  Negative
4                     Virginia Walmart shooting   Neutral
5                           LGBTQ club shooting  Negative
6              University of Virginia shooting   Neutral
7                        Raleigh spree shooting   Neutral
8                  Greenwood Park Mall shooting   Neutral
9          Highland Park July 4 parade shooting   Neutral
10                Church potluck dinner shooting   Neutral
11                     Concrete company shooting   Neutral
12                Tulsa medical center shooting   Neutral
13             Robb Elementary School massacre   Neutral
14                Buffalo supermarket massacre  Negative
15         Sacramento County church shooting   Neutral
16                  Oxford High School shooting   Neutral
17                        San Jose VTA shooting  Negative
```

There was one outlier in the data, in row 66 the sentiment was displayed as positive.
The exact statement in the data was:
“Harper-Mercer's mother said in multiple online postings that he had Asperger's syndrome. Harper-Mercer graduated from the Switzer Learning Center, a school for students with special needs, emotional difficulties, autism, and Asperger's syndrome.”

```
In [62]: row_index = 66
         print(df.iloc[row_index])
```

```
case                                           Umpqua Community College shooting
location                                                        Roseburg, Oregon
date                                                                   10/1/2015
summary                         [26-year-old, chris, harper, mercer, opened, f...
fatalities                                                                     9
injured                                                                        9
total_victims                                                                 18
location.1                                                                School
age_of_shooter                                                                26
prior_signs_mental_health_issues                                         Unclear
mental_health_details           Harper-Mercer's mother said in multiple online...
weapons_obtained_legally                                                     Yes
where_obtained                  From the home he shared with his mother. All w...
weapon_type                                                             multiple
weapon_details                  9 mm Glock pistol, .40 caliber Smith & Wesson,...
race                                                                       Other
gender                                                                      Male
type                                                                        Mass
year                                                                        2015
sentiment                                                               Positive
Name: 66, dtype: object
```

References

1. https://plotly.com/python/choropleth-maps/

2. https://scikit-learn.org/stable/modules/svm.html#:~:text=Support%20vector%20machines%20(SVMs)%20are,classification%2C%20regression%20and%20outliers%20detection.
3. https://www.kdnuggets.com/2020/04/dbscan-clustering-algorithm-machine-learning.html
4. https://scikit-learn.org/stable/modules/generated/sklearn.ensemble.IsolationForest.html