MMA 860: Acquisition and Data Management
Team: *Bremner*

To: Prof. Alex Scott

Team Project Name:
Proposal: The impact of socioeconomics conditions on gun violence

Due Date: April 16th, 2021

Team Members: Bremner

Student Name	**Student Number**

Order of files:

Filename	**Pages**	**Comments and/or Instructions**
MMA 860 Bremner Team Project Final Report.docx	21	The title is included in the page count.

Memorandum

To: A. Scott, Professor of Analytics, Smith School of Business, Toronto, ON
From: Team Bremner, Smith School of Business at Queens University, MMA 860
Date: April 16th, 2021
Re: Socioeconomics and gun violence

Executive Summary:

Does socioeconomics play an integral role in gun violence? We want to push past political correctness and explore the relationships between education level, median income, poverty level, and race with the number of shootings. Our goal is to see if there is any merit to the general idea that poor, uneducated, high minority population cities experience more gun violence.

We wanted to analyze our hypothesis that someone's socioeconomic standing may lead them to gun violence. We would expect local, state, and federal governments to address the contributing factors and reduce overall gun violence if we can prove this. These investments could include providing incentives for the affected communities' children to complete high school and join the workforce to improve median income.

Team Bremner has been hired to understand and evaluate this issue and uncover a relationship between gun violence and socioeconomic factors in the United States of America.

We found two sources of data for gun-related incidents and socioeconomics and joined them:

1. Gun violence data: Information on all the gun violence from 2013 to 2018 with a total of about 239,678 incidents and 29 attributes,
2. Socioeconomics data: Four datasets containing median household income, percentage of population below the poverty line, high school graduation percentage, and percentage of race per city in American cities in 2015 with about 30,000 records and three attributes for each dataset.

Team Bremner employed five data analysis techniques. We undertook the first three techniques to complete our analysis: data merging, data cleaning, and feature engineering. This allowed us to understand the data, structure, attributes, and completeness, resulting in a single dataset ready for modeling.

We then developed a linear regression model for our fourth activity and applied various techniques to improve model results. These techniques included adding/removing variables, dealing with outliers, and monitoring model accuracy and performance-based variable values.

For our final piece, we created an executive dashboard that further analyzes how prominent gun violence is in the United States and what may be contributing to it.

Opportunity & Recommendation:

Our objective was to build a linear regression model that predicts the number of incidents in cities based on the poverty rate, median income, ethnic background, and high school completion rate. As indicated in the equation below, our final model's coefficients are almost zero, and therefore, cannot predict the required objectives[1].

The final model was:
Average Incident Per Month Per Capita =
0.000o14 - 0.000001404 * (Percentage Completed High-School) - 0.000000000198 * (Median Income) - 0.000001716 * (Asian Share of Population) - 0.0000007791 * (Hispanic Share of Population) - 0.0000005782 * (Poverty Rate).

[1] For details of modeling approach and variable selection refer to Managerial Description's Modelling Section.

Although we were unable to conclude that our hypothesis was correct, we learned that some cities are significantly struggling with gun violence. We would recommend that the United States government investigate and intervene in the towns of Martin, Mandan, King, and Panam as this seems to be where most of the gun violence is occurring per capita. We would also recommend that the United States Government investigate the city of Chicago. In absolute terms, the city of Chicago experienced 11,583 shootings from 2013 – 2018; this is 7,638 more shootings than the second-highest city of Baltimore that had 3,945 shootings over the same time frame.

Next Steps:

We could not use this model to determine that a person's socioeconomic circumstances make them prone to gun violence. We believe several factors may have impacted the model's accuracy and predictability. Therefore we investigated further to explore potential issues.
List of issues to be considered in the next phase:

- Were the four socioeconomics datasets sufficient to test the hypothesis?
- Did datasets used in our model have insufficient predictors?
- Was linear regression the right choice for this model?
- Were there mistakes made during the data-gathering steps?
- Did we have sufficient samples for this initiative?

We have several leads to check out.

1. Data

The final dataset contained 50 variables from five different datasets: one with daily gun violence records across American cities from 2013 to 2018 and four socioeconomics datasets. The final model found only a few variables statistically significant and leads us to believe that the dataset may not contain sufficient socioeconomic factors. The gun-related incident data was collected for a different purpose, as there was detailed information about gun types, shooter and victim relationships attributes irrelevant to our hypothesis.

These datasets did not have key attributes to join, which created several challenges when merging them using "city" as a key. In the next phase, we could use text analytics to connect datasets using the URL column to scrape data from the URL sites and identify/connect cities. Text Analytics were out of scope in this initiative.

2. **Model**:

We performed extensive data cleaning and feature engineering in our previous steps, resulting in a dataset ready for modelling. It may have been more appropriate to use the "Test Test Test" (TTT) philosophy of modeling and let our data select the model with variables. This approach did not succeed due to hardware limitations, and we had to resort to our biases and feed the model variables that we believed were relevant. In the next phase, we recommend using the TTT approach as the first step in the modeling process.

We believe using a different prediction model instead of linear regression may be more appropriate. A categorical prediction of whether a shooting would occur or not based on socioeconomic indicators may be a better predictor than "Average Incident Per Month Per Capita".

Managerial Descriptions

All datasets for this project were downloaded from public sites. Data merging, data cleaning, feature engineering, and model building activities were performed in Excel and R. We used Tableau for the last activity to create a dashboard to highlight our findings and relationships between variables in our datasets.

The project details and the R code can be accessed from Team Bremner's GitHub page.

Data Merging:

The data needed to conduct our analysis on gun violence in the United States of America was spread across five datasets. Our primary dataset included information on all the gun violence in the USA from 2013 to 2018. This dataset included 29 variables and 239,678 records.

Our remaining datasets are related to socioeconomic factors like median household income, population percentage below the poverty line, high school graduation percentage, and percentage of race per city. Each of the socioeconomic datasets included three variables and 29,322 records.

The way we had decided to conduct our analysis was with our data being at the city level; all our datasets contained this level of granularity. We uncovered the main issue: there was no apparent link to join our socioeconomic datasets to our primary gun violence dataset.

The one common element between them was that our primary gun violence dataset contained the city name, state, latitude, and city's longitude. We had initially considered joining on city name; however, the city names were not consistent between each dataset, making it difficult to join on the city name. As a result, we tried to join on latitude and longitude. Given that our socioeconomic datasets had each city's name, we used the geography datatype in Excel to produce the latitude and longitude for each city. However, joining on both latitude and longitude was inconsistent. We used a left join while using our gun violence dataset as our primary dataset and the socioeconomic sets as the right. We decided to use a left join as we wanted to ensure that we included all the records from our main dataset and only wanted to include records from our socioeconomic dataset that matched. This join ultimately not working as intended, incorrectly joining the wrong socioeconomic data to the incident data. Next, we tried to join on the city name and state name. However, this required extensive cleaning to try and use this as the join criteria. Once the data had been cleaned, we joined our datasets on city name and state; this yielded favourable results. While our join was successful, we were not able to join all our data together. At the end of this process, we ended up with a dataset that contained all our original records and 213,671 records from our joined datasets.

Data Exploration & Merge	Before	After
# of tables	5	1
Variables (29 gun) + socioeconomics data variables	29||3|| 3|| 3|| 3	34
Observations	239,677	213,671
Variables with missing data	6	4
Char-type variables	20	19
# of variables dropped	N/A	7
# of new variables created	N/A	0

Table 1. Data Exploration & Merge Activity

A business-friendly data dictionary of the final dataset explains variables in non-technical terms created to meet data governance requirements.

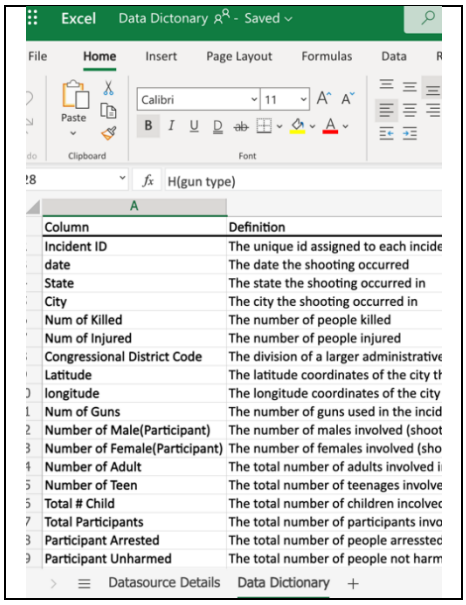


Figure 1. Data Dictionary for business users and data governance teams

Data Cleaning:

The raw data from our gun violence dataset that included information on shootings from 2013 – 2018 presented several challenges. This was not a well-curated dataset that was immediately ready for modelling. It was necessary to conduct extensive cleaning and transformation to prepare it for analysis. There were two main issues that we were faced with when cleaning our data. 1. Data that presented information about the number of people involved, age of participants, or how many people were injured/killed was not in a format that was easy to understand and interpret. Refer to figure 2. 2. Our data was not tidy and had more than one value in each cell.

To address these concerns, we first needed to understand what each column is trying to tell us and interpret the information. A small team was assigned to dissect each column to understand its contents and its contribution to the overall dataset. Once we were able to interpret each column, we leveraged the tidy data principles to ensure that each variable has its column, each observation has its row, and each value has its cell. This was a complex process, given how much data was stored in each cell. However, we could parse out all the necessary information and collect it into appropriate columns, which allowed us to build a very robust dataset.

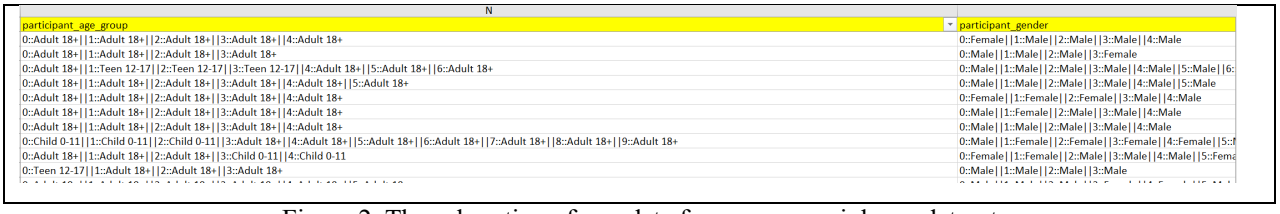


Figure 2. The subsection of raw data from our gun violence dataset

Feature Engineering:

Creating a dataset that follows the tidy data principles allowed us the flexibility to feature engineer more variables into our dataset that enhanced our analysis. The feature engineering that was applied allowed us to extrapolate further the interaction and indicator variables needed to capture every factor that our data contained.

Once we included all the necessary interaction and indicator variables in our dataset, we addressed the amount of data we lost from our first data merge. When we joined our data in step one, we lost approximately 18% of our data. This resulted from many city names being named inconsistently and our team not capturing all the necessary

changes that needed to take place in step one (refer to figure 3). To ensure that we could reduce the amount of data lost, we went back and checked which symbols, padded space, or words were causing us to lose so many records. Once we understood where we need to make changes, we could reduce the amount of lost data from 18% to 8% before aggregating. Based on further analysis of our unjoined records, we concluded that they could not be joined because the city name is missing in the file were joining. We could not capture all the errors and inconsistencies in each city name, or each city name may have a typo preventing us from joining appropriately. Once we were able to resolve this issue and limit the amount of data we lost in our join, we accurately aggregated our data at the city level to begin modelling and testing our hypothesis.

Carlisle-Rockledge CDP	Albuquerque (Los Ranchos)
Mount Olive CDP (Coosa County)	Albuquerque (Los Ranchos De Albuquerque)
Mount Olive CDP (Jefferson County)	Anderson (county)
Carmel-by-the-Sea city	Anderson County
Hartsville/Trousdale County	Batesburg (Batesburg-leesville)

Figure 3. Two datasets discrepancy for city values.

Feature Engineering results	Before	After
Table name	join_data_7	df3
Variables	34	50
Observations	240,091	240,091
Variables with missing data	4	0
Char-type variables	19	2
# of variables dropped	8	N/A
# of new variables created	N/A	24

Table 2. Feature Engineering Check List

Modeling:

Our objective was to build a linear regression model that predicts the number of incidents in cities based on the poverty rate, median income, ethnic background, and high school completion rate.

Given that we have performed extensive data cleaning and feature engineering, we created a dataset ready for modelling. To construct our model, we initially used the "Test Test Test" (TTT) approach and wanted data to play a role in selecting the model with variables that fit the data best. We expected this method would show some evidence towards proving that socioeconomic factors impact gun violence. This approach failed since our machines were unable to process our immense amount of data. We then resorted to the Average Econometric Regression (AER) approach. We had to choose which variables to feed into the model and noticed that although the dataset contained a lot of information, the variables directly related to testing the hypothesis were limited to eight. The final model had five variables with significant t-values.

Modeling Approach	Variable Used	Final # of Vars
TTT	All	N/A, TTT approach abandoned
AER	8	5

Table 3: Modelling Approach and number of variables

We had aggregated data at the city level. Still, we noticed outliers in the population-for instance, cities with a population of two, severely impacting our model's "R^2". The measure of how close the data is to the fitted regression line.

Filter	Population	Adjusted R^2	Obs. Removed	Notes
1	>50	0.0216	21	
2	>100	0. 0255	83	
3	>500	0. 0329	1,047	
4	>1,000	0. 0518	2,197	
5	>2,500	0. 0897	3,839	
6	>5,000	0. 1536	5,265	*Intercept T-test fails at (p.0.05)*
7	>10,000	0. 3378	6660	

Table 4. Outlier impact on model explainability (Adjusted R-Squared)

To address this issue, we standardized our data to a "per capita" basis to account for the significant population differences in each city. However, this did present some challenges. To capture the populations of each city, we used the geography data type that is built into Excel. However, some of the populations that were returned were incorrect. (e.g., Mandan, North Dakota returned a population of 50 via geography datatype. When we went to verify this, we learned that Mandan has a population of 22,000). This is a significant difference that strongly affected our dependent variable value and rendered our model nearly unusable.

The final model was:
Average Incident Per Month Per Capita =
0.000o14 - 0.000001404 * (Percentage Completed High-School) - 0.000000000198 * (Median Income) - 0.000001716 * (Asian Share of Population) - 0.0000007791 * (Hispanic Share of Population) - 0.0000005782 * (Poverty Rate).

Model observations	Example	Impact
Outlier examples	Cities with tiny population: 2	Severely low R-Squired values (0.004)
Data Accuracy	Dataset population for Madan = 50 Actual value = 22,000	Affects model accuracy.
Insignificant variables	Detailed meta-data gun information.	Low t-value, dropped from the model.
Insignificant variables	Detailed shooter/victim relationship	Low t-value, dropped from the model.

Table 5: Impacts of data and variables on the model.

Executive Dashboard:

Based on the data exploration that was conducted throughout our analysis. We created an executive dashboard that will take our readers on a journey to understand how gun violence is plaguing the United States. Our dashboard outlines the macro gun violence trends occurring in the USA from 2013 – 2014. Our analysis then explores which gender is more likely to be involved in an incident around gun violence. Finally, our analysis determines how an individual's socioeconomic circumstances may lead them to gun violence. Our dashboard can be found in the packaged Tableau file.

Technical Descriptions:

In data science and analytics projects, it's critical to ensure that all aspects of each activity are appropriately documented and code can be run with the same results. Our project had five different activities performed by groups of two or three, so we ensured that each activity is easily handed off to the teams responsible for the next activity. We combined each activity's R code into one code. We also created unique sections in R and sequentially labeled them so technical users can quickly identify each section, as shown in Figure 4.

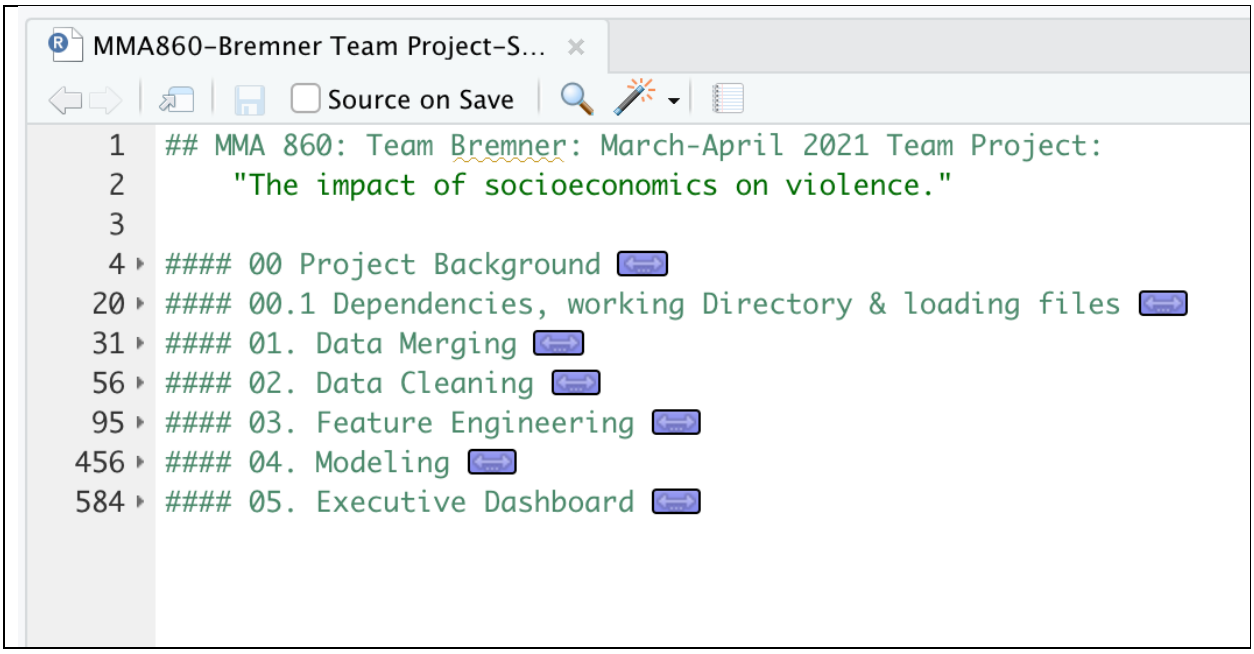


Figure 4: R-code is created to cross-reference each activity for technical users.

We also included ample comments in the R code for technical users, as shown in figure 5.

```
44
45 # Note about datafile: The original gun-violence-data_01-2013_03-2018.csv file was saved as
46 #    "gun violence base data_original.xlsx" Excel and
47 # all socioecinomics data were added to this file in different sheets.
48 #  The "gun violence base data_original.xlsx" is used as the master data file.
49 gun_data <- read_excel("gun violence base data_original.xlsx",
50                        sheet = "gun violence base data")
51 demographic_data <- read_excel("gun violence base data_original.xlsx",
52                                sheet = "demographics")
53
54 head(demographic_data)
55 str(demographic_data)
56
57 #creating our join
58 #join_data_2 did not work as expected, created over 25 million records
59 join_data_2 <- left_join(gun_data,demographic_data,by=c("latitude", "longitude"))
```

Figure 5. R code accompanied with proper comments for a technical description.

Data Merging:

The data exploration and merge in our R code is section 2 labeled "*#### 02. Data join & merge ####*" A total of seven different attempts were made to join gun incident data with four socioeconomics data. For each attempt, results were compared, and if they were not satisfactory, we tried a different approach until we were satisfied with the result.

The initial gun violence data "gun-violence-data_01-2013_03-2018 2.csv" was re-named and saved as "gun violence base data.xlsx." We then copied the socioeconomics data in different sheets listed below:

Original File	Converted to Excel	Sheet Name	Status
gun-violence-data_01-2013_03-2018 2.csv	gun violence base data.xlsx	gun violence base data	In-use
MedianHouseholdIncome2015.csv	gun violence base data.xlsx	median income	In-use
PercentagePeopleBelowPovertyLevel.csv	gun violence base data.xlsx	poverty	In-use
PercentOver25CompletedHighSchool.csv	gun violence base data.xlsx	High school completion	In-use

PoliceKillingsUS.csv	gun violence base data.xlsx	police killing	N/A
ShareRaceByCity.csv	gun violence base data.xlsx	demographics	In-use

Table 6: File sources Original and converted.

```
#reading in our datafile
library(readxl)
library(tidyverse)
library(dplyr)

gun_data <- read_excel("C:\\Users\\derek\\OneDrive\\Desktop\\gun violence\\gun violence base data.xlsx", sheet=1)
demographic_data <- read_excel("C:\\Users\\derek\\OneDrive\\Desktop\\gun violence\\gun violence base data.xlsx", sheet=2)

head(demographic_data)
str(demographic_data)
#creating our join
join_data_2 <- left_join(gun_data,demographic_data,by=c("latitude", "longitude"))

join_data_3 <- left_join(gun_data,demographic_data,by=c("city","state"))

join_data_4 <- left_join(gun_data,demographic_data,by="latitude")

join_data_5 <- left_join(gun_data,demographic_data,by="latitude")

join_data_6 <- left_join(gun_data,demographic_data,by=c("latitude", "longitude"))

join_data_7 <- left_join(gun_data,demographic_data,by=c("city","state"))
md.pattern(join_data_3)

install.packages("xlsx")
library(xlsx)
#exporting joined file to excel
write.xlsx(join_data_7, file="mma860.xlsx",sheetName="Base Data", append = FALSE)

library(rio)
export(join_data_7,"MMA 860 Project.xlsx")

memory.limit(size=80000)
```

Figure 6. Exploring & Merging data

Data Cleaning:

The data cleaning process in our R code coded step-by-step process in the third section, labeled "#### 02. Data Join & Merge ####". There are 16 sequential steps, and each step is labeled with a brief description. Below is a high-level technical overview of the Feature Engineering process:

1. Used the output data from previous activity as a starting point.
2. Removed some variables that cannot be used in our model:
    - Address: our model will only focus on the city scop, not on the street scop
    - Participant_age: we already have the participant_age_group column and will only focus on the age group instead of the specific ages.
    - Participabnt_name: we do not care about the individual participant's name.
    - City_and_state: we have city and state columns separately.
    - State_id: we already have a state column.
    - Latitude. y: Redundant column
    - longitude. y: Redundant column.
3. Converted median_income, poverty_rate and high_school_completion_rate type to numeric.
4. Split 'participant_gender' into 'Number of Male (Participant)' and 'Number of Female (Participant),' then change NA to 0.

```
# 4. Crete column called  # of Male of Participant,# of Female of Participant####
df2 <- df
str(df2)

df2$"Number of Male(Participant)" <- str_count(df2$participant_gender , "Male")
df2$"Number of Female(Participant)" <- str_count(df2$participant_gender , "Female")
#change NA to 0
df2$`Number of Male(Participant)`[is.na(df2$`Number of Male(Participant)`)] <- 0
df2$`Number of Female(Participant)`[is.na(df2$`Number of Female(Participant)`)] <- 0
sum(is.na(df2$`Number of Female(Participant)`))
sum(is.na(df2$`Number of Male(Participant)`))
```

Figure 7: split process snapshot

The content of Participant gender's before and after feature engineering is shown in figure 7.

participant_gender	Number of Male(Participant)	Number of Female(Participant)
0::Male||1::Male||3::Male||4::Female	3	1
0::Male	1	0
0::Male||1::Male||2::Male||3::Male||4::Male	5	0
0::Female||1::Male||2::Male||3::Male	3	1
0::Female||1::Male||2::Male||3::Female	2	2
0::Female||1::Female||2::Female||3::Female||4::Male||...	2	4
0::Male||1::Female||2::Male||3::Female||4::Female||5::...	3	3
0::Male||1::Male||2::Male||3::Male||4::Male	5	0
0::Male||1::Male||2::Male||3::Male||4::Male	5	0
0::Male	1	0
0::Male||1::Male||2::Male||3::Male	4	0
0::Male		
0::Male||1::Male||2::Male||3::Male||4::Male		
0::Female||1::Female||2::Male||3::Male||4::Male||5::Male		
0::Male||1::Male||2::Male||3::Male||4::Male		
0::Female||1::Female||2::Female||3::Female||4::Femal...		
0::Male||1::Male||2::Female||4::Male||5::Male||6::Male		

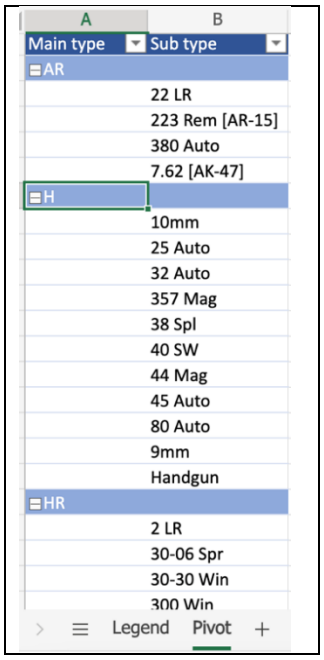
Figure 8. Feature engineering gender from participant_gender variable

5. Split 'participant_age_group' into 'Number of Adults,' 'Number of Teen,' and 'Total # Child,' then change NA to 0 as shown in figure 9.

participant_age_group	Number of Adult	Number of Teen	Total # Child
0::Adult 18+||1::Adult 18+||2::Adult 18+||3::Adult 1...	5	0	0
0::Adult 18+||1::Adult 18+||2::Adult 18+||3::Adult 18+	4	0	0
0::Adult 18+||1::Adult 18+||2::Adult 18+||3::Adult 1...	5	0	0
0::Adult 18+||1::Adult 18+||2::Adult 18+||3::Adult 18+	4	0	0
0::Adult 18+||1::Adult 18+||2::Teen 12-17||3::Adult ...	3	1	0
0::Adult 18+||1::Adult 18+||2::Adult 18+||3::Adult 1...	6	0	0
0::Adult 18+||1::Adult 18+||2::Child 0-11||3::Child 0...	2	1	3
NA	0	0	0
0::Teen 12-17||1::Teen 12-17||2::Teen 12-17||4::Ad...	1	3	0
0::Teen 12-17||1::Adult 18+||2::Adult 18+||3::Adult ...	6	1	0
0::Adult 18+	1	0	0
0::Adult 18+	1	0	0
0::Adult 18+||1::Adult 18+||2::Adult 18+||3::Adult 1...			

Figure 9. Feature engineering age factors from participant_age_group variable

6. Created the "total # Participant" by coding to get the max of total genders and the total number of age groups.
7. Steps 4 and 5 were repeated to deal with 'Participant Type', 'Gun types', 'gun_stolen' and 'participant_relationship'
8. We also consolidated 13 different gun types into 8 types: ‘Unknown(gun type)’, ’ SG(gun type)’, ’ Other(gun type)’, ’ H(gun type)’, ‘HR(gun type)’, ’Hunting Rifle(gun type)’ and ‘AR(gun type)’
   Note: A separate data-file 'gun legend' was created to map these fields.



A	B
Main type	Sub type
AR	
	22 LR
	223 Rem [AR-15]
	380 Auto
	7.62 [AK-47]
H	
	10mm
	25 Auto
	32 Auto
	357 Mag
	38 Spl
	40 SW
	44 Mag
	45 Auto
	80 Auto
	9mm
	Handgun
HR	
	2 LR
	30-06 Spr
	30-30 Win
	300 Win

Legend Pivot +

Figure 10: Excel was used to create 'gun legend' and map gun types.

9. We also performed clean-up on several columns and removed irrelevant columns.
10. Several column names were changed to meaningful names for the modeling.

```
#9. Rename Columns####
names(df3)[names(df3) == 'incident_id'] <- 'Incident ID'
names(df3)[names(df3) == 'state'] <- 'State'
names(df3)[names(df3) == 'city'] <- 'City'
names(df3)[names(df3) == 'n_killed'] <- 'Num of Killed'
names(df3)[names(df3) == 'n_injured'] <- 'Num of Injured'
names(df3)[names(df3) == 'congressional_district'] <- 'Congressional District Code'
names(df3)[names(df3) == 'latitude.x'] <- 'Latitude'
names(df3)[names(df3) == 'longitude.x '] <- 'Longitude'
names(df3)[names(df3) == 'n_guns_involved'] <- 'Num of Guns'
names(df3)[names(df3) == 'poverty_rate'] <- 'Poverty Rate'
names(df3)[names(df3) == 'median_income'] <- 'Median Income'
names(df3)[names(df3) == 'Total # Child '] <- 'Number of Children'
names(df3)[names(df3) == 'Total # Participant'] <- 'Total Participants'
str(df3)
summary(df3)
```

Figure 11. Standardizing variable names for modeling exercise

11. Missing values: The majority of missing values were from the demographic data. We used the multiple imputation method to address the issue.

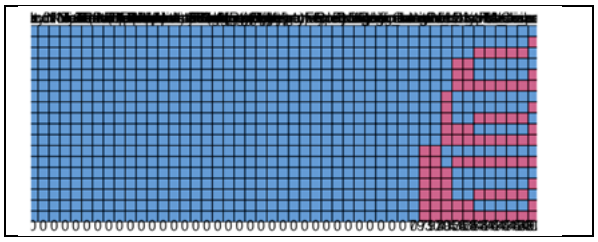


Figure 12. Identifying variables with missing data

12. Summarize statistical analysis

```
#11. Summary Statistics
mean(df3$`Num of Killed`, na.rm=TRUE)
mean(df3$`Num of Injured`, na.rm=TRUE)
mean(df3$`Number of Male(Participant)`, na.rm=TRUE)
mean(df3$`Number of Female(Participant)`, na.rm=TRUE)
mean(df3$`Num of Guns`, na.rm=TRUE)
mean(df3$`Total Participants`, na.rm=TRUE)

sd(df3$`Num of Killed`, na.rm=TRUE)
sd(df3$`Num of Injured`, na.rm=TRUE)
sd(df3$`Number of Male(Participant)`, na.rm=TRUE)
sd(df3$`Number of Female(Participant)`, na.rm=TRUE)
sd(df3$`Num of Guns`, na.rm=TRUE)
sd(df3$`Total Participants`, na.rm=TRUE)
```

Figure 13: Prepared data and performed statistical analysis variable.

13. Bivariant Analysis

```
#12. Bivariate analysis####
cor(df3$`Total Participants`, df3$`Median Income`, use="complete.obs")
cor(df3$`Total Participants`, df3$`Poverty Rate`, use="complete.obs")
cor(df3$`Total Participants`, df3$`Massive Shooting (Y/N)`, use="complete.obs")

cor(df3[c('Total Participants', 'Massive Shooting (Y/N)', 'Participant Injured','Num of Guns')], use="complete.obs")
```

Figure 14. Bivariant Analysis

```
> cor(df3[c('Total Participants', 'Massive Shooting (Y/N)', 'Participant Injured','Num of Guns')], use="complete.obs")
                     Total Participants Massive Shooting (Y/N) Participant Injured  Num of Guns
Total Participants           1.000000000           0.256852136          0.34984654  0.003302272
Massive Shooting (Y/N)       0.256852136           1.000000000          0.39400737 -0.002999669
Participant Injured          0.349846535           0.394007368          1.00000000 -0.041317862
Num of Guns                  0.003302272          -0.002999669         -0.04131786  1.000000000
>
```

Figure 15. Colinear analysis

14. We also performed basic visualizations using R and Tableau. A few examples are shown below:

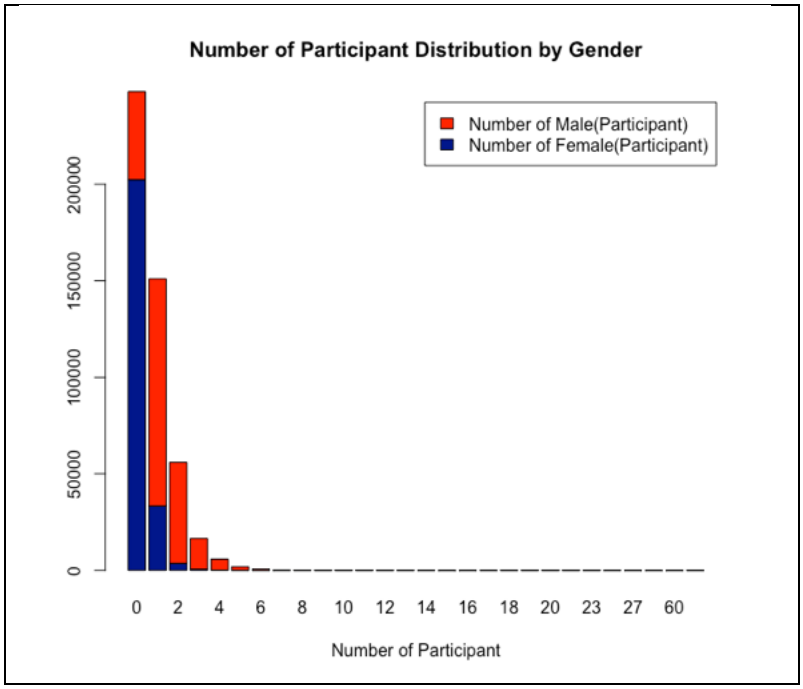


Figure 16. Distribution of Participants by Gender

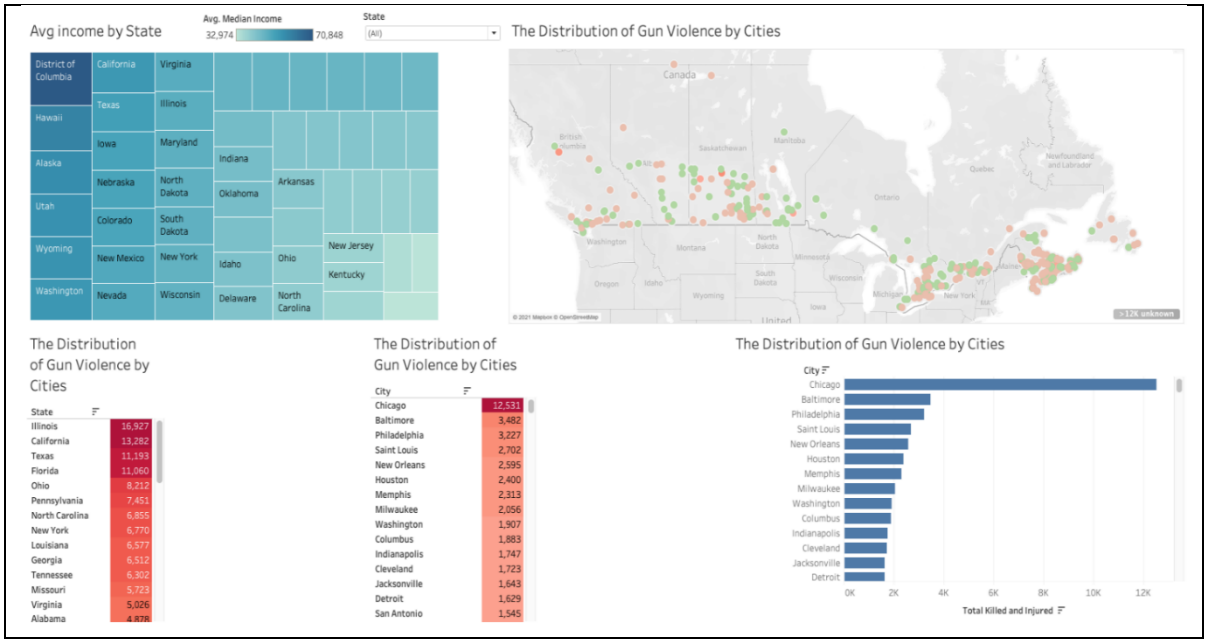


Figure 17. Distribution of Gun incidents by cities

Figure 18. Socioeconomics visualization: poverty rate distribution

15. The final step was to store the modified dataset and pass it to the feature engineering team

Feature Engineering:

The feature engineering process code is under #### 03. Feature Engineering #### in the R code. There were two significant activities associated with the feature engineering task.

3. Address the missing geographical data.
4. Aggregate data to the appropriate level for the model. (Aggregate from daily incidents to weekly and monthly level based on the city)

These tasks used the following datasets:

- Task2
- MedianHouseholdIncome2015.csv
- PercentagePeopleBelowPovertyLevel.csv
- PercentOver25CompletedHighSchool.csv
- ShareRaceByCity.csv
- State.xlsx
- US Cities.xlsx

The final output of this task was one dataset that contains data aggregated to weekly and monthly levels for each city.

Data Engineering Part One- Geo Data:
"City", the key that connects datasets, proved to be a significant challenge in the feature engineering process. We realized several challenges impeded efforts to merge the datasets based on cities. These challenges included observations with missing city values, inconsistencies in the city names, including typos, city and state combined, and usage of characters like "/", "()" and "-" that prevented us from joining cities appropriately.

Carlisle-Rockledge CDP	Albuquerque (Los Ranchos)
Mount Olive CDP (Coosa County)	Albuquerque (Los Ranchos De Albuquerque)
Mount Olive CDP (Jefferson County)	Anderson (county)
Carmel-by-the-Sea city	Anderson County
Hartsville/Trousdale County	Batesburg (Batesburg-leesville)

Figure 19. Challenges of using the city as our key.

We had to go back to the meta-data and investigate why the cities could not be matched and remedy the issues one step at a time as listed below:

- Check specific symbols, stop words, upper or lowercase, and extra spaces at the start, middle, or end of each field
- Check the length of the city in each dataset and address padded values (e.g., last characters padded by space)

Technical step by code:

1. Join state data & reduce missing values:
   Create table "*Task_abb*", convert the state abbreviation to uppercase, and add to "Task2."

```
#keep all uppercase
Task2$State<-toupper(Task2$State)
state$State<-toupper(state$State)

#join state abbreviation by state dataset
Task_abb<-left_join(Task2,state,by=c("State"))
```

Figure 20. Convert to uppercase to address mis-cased values

   Note: *state.abb[match(Task2$State,state.name)]* caused missing values. To address this, we imported the full state and state abbreviation values. State_abb now has no missing values.

2. Clean "City" column:
   - Clean City column in both "*joined table*" & "*Task2*".
   - Remove special characters (e.g. backslash)

- Look for special acronyms like "st." (Saint)
- We removed unnecessary words like "city"," CDP", "country", and "town"

```
#remove the bracket string
Task_abb$City<-gsub("\\(.+?\\)", "", Task2$City)

#change some specific word to match the city
#only can select the large number of missing by manually
Task_abb$City<-sub(Task_abb$City, pattern = "Saint", replacement = "St.")
Task_abb$City<-sub(Task_abb$City, pattern = "Bronx", replacement = "Bronxville")
Task_abb$City<-sub(Task_abb$City, pattern = "Chesterfield", replacement = "Chester")
#Task_abb$City<-sub(Task_abb$City, pattern = "Wilkes Barre", replacement = "Wilkes-Barre")
#Task_abb$City<-sub(Task_abb$City, pattern = "Winston Salem", replacement = "Winston-Salem")

#ignore the word
stopwords = c("City","city","Country","country","CDP","town","village","Town","Village","borough","and",
              "municipality","-Clarke County unified government (balance)","-Richmond County consolidated government (balance)")
#remove specific words
Task_abb$City<-removeWords(Task_abb$City,stopwords)
```

Figure 21. Capturing proper "city" value

3. Since city names are not unique in the US, we needed to merge "state_abb" and "city"

```
#merge the state and city to unique
Task_abb$state_all<- with(Task_abb, paste0(State_abb,sep = " ", City))
#remove the space
Task_abb$state_all<-gsub(" ", "", Task_abb$state_all, fixed = TRUE)
Task_abb$state_all <- str_replace_all(Task_abb$state_all, fixed(" "), "")
#upper case and remove the space
Task_abb$state_all<-toupper(Task_abb$state_all)
```

Figure 22: address same city names in different states

4. Import geographic data and clean the city column. This step includes the following actions:
   Remove stop-words
   Remove brackets
   Remove "/"
   Change "-" to space
   Remove all spaces

```
#remove stopwords,remove string after/,remove space,remove bracket,change "-" to space
share$City<-removeWords(share$City,stopwords)
share$City<-gsub("\\(.*?\\)", "", share$City)
share$City<-gsub('-',' ',share$City)
share$City<-str_replace(share$City, '(.+)/.+', '\\1')
#share$City<-str_replace(share$City, '(.+)-.+', '\\1')
share$City<- str_replace_all(share$City, fixed(" "), "")
```

Figure 23: Import geographic data and perform data cleansing steps

5. Steps 1 to 4 are repeated for the four datasets

6. Join socioeconomics datasets (Income, poverty level, high school completed and share race) by 'Geographic area' and 'city' as keys, using full-join as we need all data

```
#need to full join not left join
join1<-full_join(High, Income, by=c("Geographic.Area","City"))
join2<-full_join(join1, share, by=c("Geographic.Area"="Geographic.area","City"))
join3<-full_join(join2, Poverty, by=c("Geographic.Area","City"))
```

Figure 24: joined socioeconomics datasets

7. Merge state and city in joined geo table named "join" to keep the city unique in different states for task2 and joined the geo table.
   Remove blanks by checking the length of strings,
   Convert values to uppercase and

Remove duplicate data.

```
join$state_all<- with(join, paste0(join$Geographic.Area,sep = " ", City))
join$state_all<-gsub(" ", "", join$state_all, fixed = TRUE)
join$state_all_geo<-toupper(join$state_all)
summary(join)
length(join3$state_all)
#duplication
join<-join[!duplicated(join$state_all_geo), ]
```

Figure 25. Merge state and city

8. Left join in Task2 and join3 and replace the space and change the type of variables.

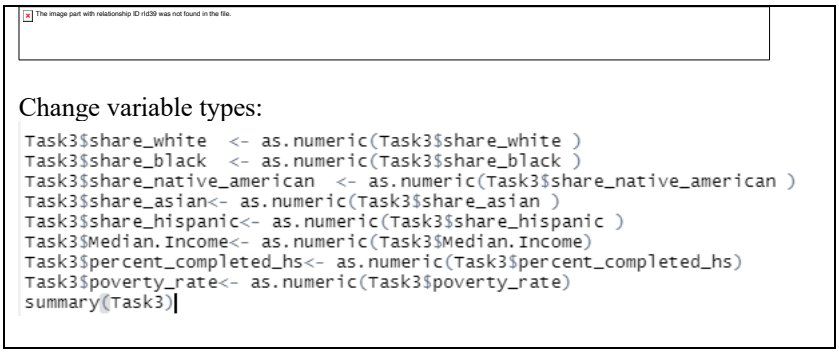


Figure 26. join task2 and join3 and change variable types

Data engineering result on joining data

Data Engineering	Before	After	Reason
Missing geodata	18%	8%	Data Engineering efforts

Table 7. Data Engineering improved data merge by 10%

Despite 10% improvement, there is still a 30% that cannot be merged because of missing city values. Although 30% missing values seem high, it is within an acceptable level. According to Principled Missing Data for Researchers, "…*Missing data are a rule rather than an exception in quantitative research. Enders (2003) stated that a missing rate of 15% to 20% was common in educational and psychological studies…*[2]"

Even after dropping 25%-30% of our data, we still have over 10,000 non-missing observations, plenty for a regression model.

Data Engineering Part Two-Aggregation:
To aggregate data to a weekly or monthly level, the following steps were taken:

1. create column week and year based on the date

```
Task3$Week <- week(Task3$date)
Task3$Month <- month(Task3$date)
Task3$Year <- substring(Task3$date,1,4)
```

Table 8. Aggregate date to weekly, monthly, or annually

2. Count "*Incident ID*" and sum rest of columns excluding geo data and group by week, year, state, city using SQL in R

[2] **Principled missing data methods for researchers**

```
# Weekly or monthly
Weekly<-sqldf('SELECT distinct Year,Month,Week,State,`City.x` as City,count(distinct `Incident ID`) as Number_Incident,
     sum( `Num of Injured`) as Number_Injured,Latitude,`longitude.x` as longitude,
     ...
            sum( `Home Invasion`) as Num_Home_Invasion
            , percent_completed_hs, `Median.Income`,share_white,share_black,
            share_native_american,share_asian,share_hispanic,poverty_rate

            from Task3 where Year <>"2013" GROUP BY Year,Month,Week,State,City')
```

Figure 27. Use sqldf command to prepare data for aggregation

```
# Cities:

Cities<-sqldf('SELECT distinct State,`City.x` as City,count(distinct `Incident ID`) as Number_Incident,
     sum( `Num of Killed`) as Number_Killed,
     sum( `Num of Injured`) as Number_Injured,Latitude,`longitude.x` as longitude,
     ...
            sum( `Home Invasion`) as Num_Home_Invasion

            , percent_completed_hs, `Median.Income`,share_white,share_black,
            share_native_american,share_asian,share_hispanic,poverty_rate,population

            from Task3 where Year <>"2013" GROUP BY State,City
              ')
```

Figure 28. Use "sqldf" command to prepare data for aggregation and group by city

Note:
After checking the frequency, we found the 2013 data is too small and decided to drop 2013 data all together

3. Merge state and city,
   Remove blank spaces
   Check missing data and duplicates
   Check cities that are missing in geodata
   and finally, export data back to Excel

```
#Merge state and city, Remove all the space
Create the new columns - Avg_Incident_Per_Month & Avg_Incident_Per_Month_Per_Capita
Cities$State_City<- with(Cities, paste0(Cities$State,sep = "_", Cities$City))
Cities$Avg_Incident_Per_Month<- Cities1$Number_Incident/51
Cities$Avg_Incident_Per_Month_Per_Capita<-ifelse(Cities$Avg_Incident_Per_Month != 0,
                          Cities$Avg_Incident_Per_Month / Cities$population, 0)


# Some check about missing and frequency.
check<-sqldf('SELECT * from share where City ="Louisville"  ')
check<-sqldf('SELECT *from Task3 where `City.x` ="Louisville"')
#check the freq
freq<-sqldf('SELECT distinct Year,Month, count(distinct City) as City
   from Monthly GROUP BY Year,Month')

freq1<-sqldf('SELECT distinct Year,Month, sum( Number_Incident) as Number_Incident
   from Monthly GROUP BY Year,Month')

freq1<-sqldf('SELECT *
   from v3 where Population>=100')

# Note: need to check specific cities that are missing geodata
# Export data to excel
write_xlsx(Cities,"Task3-aggregate to City_v2.xlsx")

write.xlsx(freq,"freq.xlsx",sheetName = "City Freq", append = FALSE)
write.xlsx(freq1,"freq.xlsx",sheetName = "Number_Incident Freq", append = TRUE)
```

Figure 29. Merge state, city, perform a final check, and export data to Excel

Modeling:

The modeling objective was to build and test a linear regression model in R. We used "*average monthly incidents per capita*" as the dependent variable and our socioeconomic data points as our independent variables.
The step-by-step modeling process can be found in the "#### 04. Modeling ####" section of the code

Modelling Process:

1. Data split into train and test sets: The file from data engineering activity is read and a high-level analysis is performed before data was split into 85% for train and 25% for test.

```
#Reading the datafile
df <- read_excel(file.choose())
head(df)
summary(df)
#Removing the missing values from the dataset
df2 <- df[complete.cases(df), ]
#Creating Training and Testing datasets for the
# model in 85:15 proportions of data respectively
sample <- sample.int(n = nrow(df2),
                     size = floor(.85*nrow(df2)), replace = F)
train <- df2[sample, ]
test <- df2[-sample, ]
head(train)
```

Figure 30. Modeling step 1, read & split data into train and test sets.

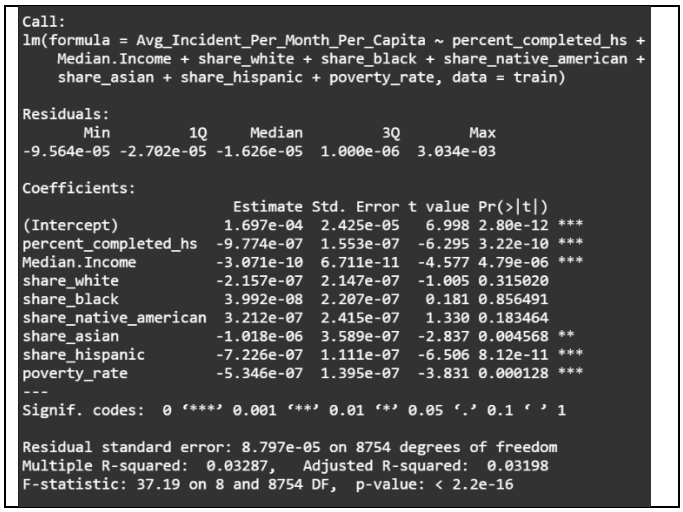
2. use lm() to predict average incidents per month and capita:

```
#Model 1 including the variables from the dataset. Dependent Variable
#  here is  to predict Average incidents that happen per month per capita.
# Independent variables are socio economic variables.
reg1 <- lm(Avg_Incident_Per_Month_Per_Capita ~  percent_completed_hs +
 Median.Income + share_white + share_black + share_native_american +
 share_asian + share_hispanic + poverty_rate, train)
summary(reg1)
plot(reg1)
ols_plot_cooksd_bar(reg1)
```

Figure 31. The first attempt in creating the linear regression model.

3. As shown in figure 18, the regression model's output shows that our model has overall predictive power as it passed the F-Test. We also noticed that several variables that need investigation because of t-test results. The variables: "share_white", "share_black", "share_native_american." were not jointly significant. This was the first sign that our hypothesis may not hold up as we thought data points such as "share_black" would be significant.

   We also noticed that the model could not explain most of the variation in our dependent variable since the "Adjusted R square' value is 0.03198, which is incredibly low.



```
Call:
lm(formula = Avg_Incident_Per_Month_Per_Capita ~ percent_completed_hs +
    Median.Income + share_white + share_black + share_native_american +
    share_asian + share_hispanic + poverty_rate, data = train)

Residuals:
       Min         1Q     Median         3Q        Max
-9.564e-05 -2.702e-05 -1.626e-05  1.000e-06  3.034e-03

Coefficients:
                        Estimate Std. Error t value Pr(>|t|)
(Intercept)            1.697e-04  2.425e-05   6.998 2.80e-12 ***
percent_completed_hs  -9.774e-07  1.553e-07  -6.295 3.22e-10 ***
Median.Income         -3.071e-10  6.711e-11  -4.577 4.79e-06 ***
share_white           -2.157e-07  2.147e-07  -1.005 0.315020
share_black            3.992e-08  2.207e-07   0.181 0.856491
share_native_american  3.212e-07  2.415e-07   1.330 0.183464
share_asian           -1.018e-06  3.589e-07  -2.837 0.004568 **
share_hispanic        -7.226e-07  1.111e-07  -6.506 8.12e-11 ***
poverty_rate          -5.346e-07  1.395e-07  -3.831 0.000128 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 8.797e-05 on 8754 degrees of freedom
Multiple R-squared:  0.03287,   Adjusted R-squared:  0.03198
F-statistic: 37.19 on 8 and 8754 DF,  p-value: < 2.2e-16
```

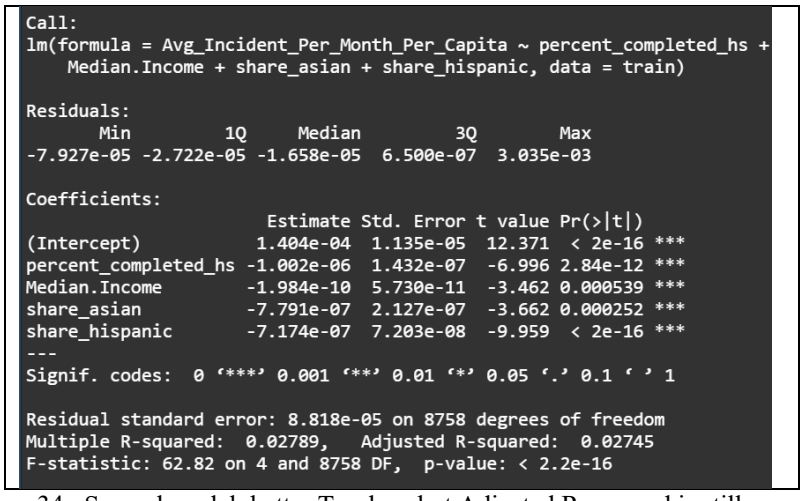
Figure 32. The output of the first regression model.

4. The second model: In our second attempt, we removed insignificant variables and limited the model to the "percentage of high-school completed," "median income," "ratio of Asian and Hispanic minorities.

5. The result of this model was a bit promising since it included jointly significant variables. However, our R-squared was still low, which was a problem. To address this, we investigated extreme low and high values (outliers) in the model, and it found out several outliers in the dataset, as shown in figure 21.

```
reg2 <- lm(Avg_Incident_Per_Month_Per_Capita ~ percent_completed_hs +
 Median.Income + share_asian + share_hispanic, train)
summary(reg2)
par(mfrow=c(2,2)) #0.02745
plot(reg2) #769 478 4294 6469
ols_plot_cooksd_bar(reg2)
```

Figure 33: The second model with insignificant variables removed.



```
Call:
lm(formula = Avg_Incident_Per_Month_Per_Capita ~ percent_completed_hs +
    Median.Income + share_asian + share_hispanic, data = train)

Residuals:
       Min         1Q     Median         3Q        Max
-7.927e-05 -2.722e-05 -1.658e-05  6.500e-07  3.035e-03

Coefficients:
                       Estimate Std. Error t value Pr(>|t|)
(Intercept)           1.404e-04  1.135e-05  12.371  < 2e-16 ***
percent_completed_hs -1.002e-06  1.432e-07  -6.996 2.84e-12 ***
Median.Income        -1.984e-10  5.730e-11  -3.462 0.000539 ***
share_asian          -7.791e-07  2.127e-07  -3.662 0.000252 ***
share_hispanic       -7.174e-07  7.203e-08  -9.959  < 2e-16 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 8.818e-05 on 8758 degrees of freedom
Multiple R-squared:  0.02789,   Adjusted R-squared:  0.02745
F-statistic: 62.82 on 4 and 8758 DF,  p-value: < 2.2e-16
```

Figure 34: Second model: better T-values but Adjusted R-squared is still very low

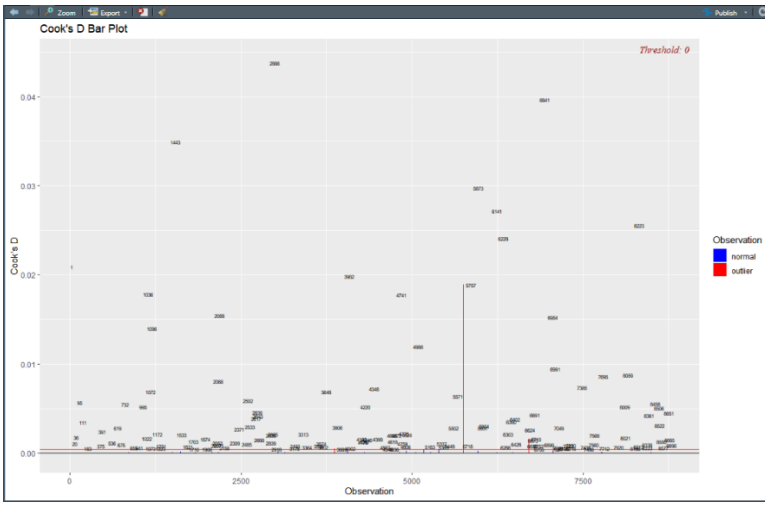


Figure 35. Cook's plot showing outliers impact on the model's prediction.

6. We also tested for heteroskedasticity.
   The heteroskedasticity hypothesis couldn't be rejected, so we needed to address this issue as well.

```
#Testing heteroskedasticity for the model reg2
    ncvTest(reg2)

Non-constant Variance Score Test
Variance formula: ~ fitted.values
Chisquare = 414.5089, Df = 1, p = < 2.22e-16
```

Figure 36. Testing for heteroskedasticity in our model.

7. Addressing low R-Squared values.

   Our initial model had a meager adjusted R-squared value of 0.02. When we filtered out lower population cities, we saw incremental improvement in our adjusted R-squared as we filtered cities out with less than 50, 100, 500, 1000, 5000, and 10000 total population. We used the Excel geography data type to retrieve population statistics for all our cities, and some of the values returned were incorrect. (e.g., Mandan, North Dakota returned a population of 50 via Excel, but a quick google search revealed a population over 22,000) This massive and unpredictable inaccuracy strongly affected our dependent variable value and rendered our R-squared nearly unusable. This issue seemed to happen more often with smaller cities with populations under 1000, which led us to filter out low population cities and observe R-squared improvements up to nearly 0.4 from a measly 0.01.

Filter	Population	Adjusted R^2	Obs. Removed
1	>50	0.0216	21
2	>100	0.0255	83
3	>500	0.0329	1,047
4	>1,000	0.0518	2,197
5	>2,500	0.0897	3,839
6	>5,000	0.1536	5,265
7	>10,000	0.3378	10,531

Table 9. Outlier impact on model explainability (Adjusted R-Squared)

Executive Dashboard:

Having the executive dashboard created in Tableau provided us tremendous flexibility in presenting our information. Our dashboard was built upon simple calculated fields that allowed us to conduct a deeper analysis of our data. The fields that we were able to create from our feature engineering and calculated fields allowed us to tell a compelling story of how gun violence has affected the United States. It also allowed us to address how individuals' socioeconomic circumstances impact their decisions to resort to gun violence. To view our analysis, please refer to the Tableau file we have submitted.