Road Incidents, Car Wreck Fatalities & Covid in the United States 2019-2020

Katie Chen & Heather Lemon

2022/05/23

Introduction:

The COVID-19 pandemic first reached the United States in January of 2020, and rapidly spread throughout the country until it reached its first peak in March of 2020. In this project, we aim to find out if the number of national road incidents and car wreck fatalities changed between 2019 and 2020 when the pandemic began. Additionally, we want to know if the COVID-19 pandemic was a factor in affecting the number of driving incidents in 2020.

Authors:

Heather Lemon - Part 1 Comparing FARS 2019 with FARS 2020 Data Katie Chen - Part 2 Comparing FARS 2020 Data with the Covid Tracking Project

Data Sets:

https://www.nhtsa.gov/file-downloads?p=nhtsa/downloads/FARS/
https://covidtracking.com/data/national

Loading in all the necessary libraries:

```
knitr::opts_chunk$set(echo = TRUE)
library(tidyverse)

## -- Attaching packages --------------------------------------- tidyverse
1.3.1 --

## v ggplot2 3.3.5     v purrr   0.3.4
## v tibble  3.1.6     v dplyr   1.0.8
## v tidyr   1.2.0     v stringr 1.4.0
## v readr   2.1.2     v forcats 0.5.1

## -- Conflicts ------------------------------------------
tidyverse_conflicts() --
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()    masks stats::lag()

library(dplyr)
library(ggplot2)
library(reshape2)
```

```
##
## Attaching package: 'reshape2'

## The following object is masked from 'package:tidyr':
##
##     smiths

library(lubridate)

##
## Attaching package: 'lubridate'

## The following objects are masked from 'package:base':
##
##     date, intersect, setdiff, union

library(glmnet)

## Loading required package: Matrix

##
## Attaching package: 'Matrix'

## The following objects are masked from 'package:tidyr':
##
##     expand, pack, unpack

## Loaded glmnet 4.1-4
```

The questions we would like to answer:

Do you think the car wreck fatality rate went up or down from 2019-2020? Did Covid have an impact on car wreck fatalities?

Part 1: FARS (Fatality Analysis Reporting System) 2019-2020 Comparisons

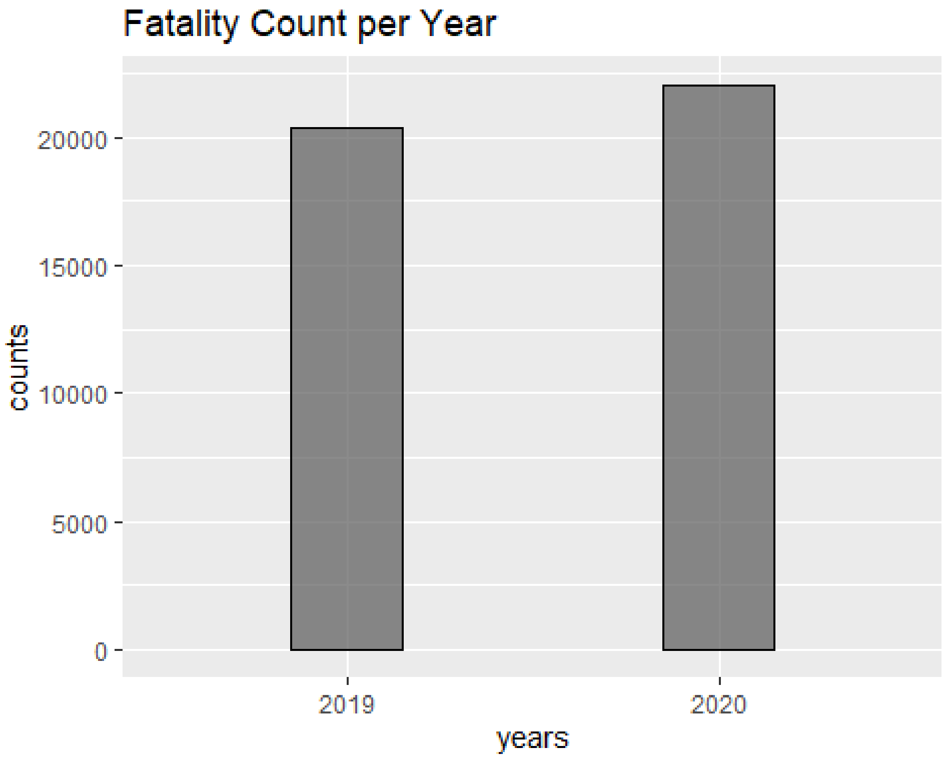
Reading in the data:

Load people dataset:

```
person2020 = read.csv("./person2020.csv")
dat20=data.frame(person2020)


person2019 = read.csv("./person2019.csv")
dat19=data.frame(person2019)
```

Total Car Wreck Deaths in 2020&2019:

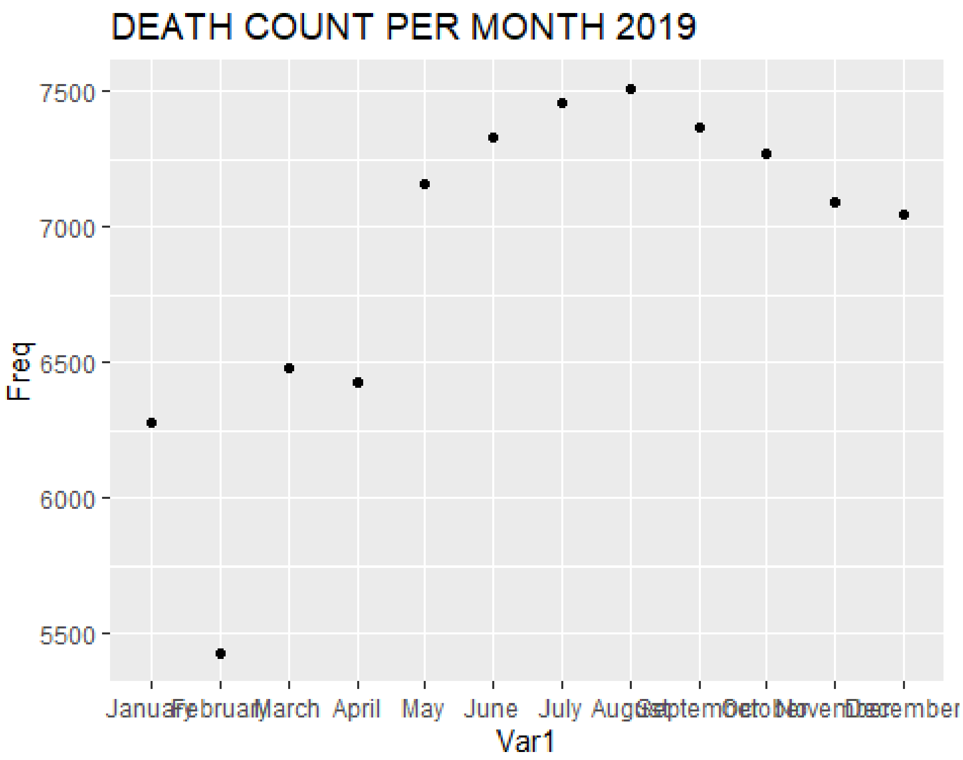
```
table_death_count2020=table(dat20$DOA)

total2020=table_death_count2020[2]+table_death_count2020[3]

table_death_count2019=table(dat19$DOA)
total2019=table_death_count2019[2]+table_death_count2019[3]

df2019=data.frame(total2019)
df2020=data.frame(total2020)

counts= c(df2019$total2019, df2020$total2020)
years = c('2019', '2020')

df=data.frame(years, counts)

ggplot(data=df, aes(x=years, y=counts))+geom_bar(stat = "identity", width =
.3,
alpha=0.7, color="black")+ggtitle("Fatality Count per Year")
```



```
percent_increase = ((counts[2]-counts[1])/counts[1]) * 100
percent_increase

## [1] 8.10333
```

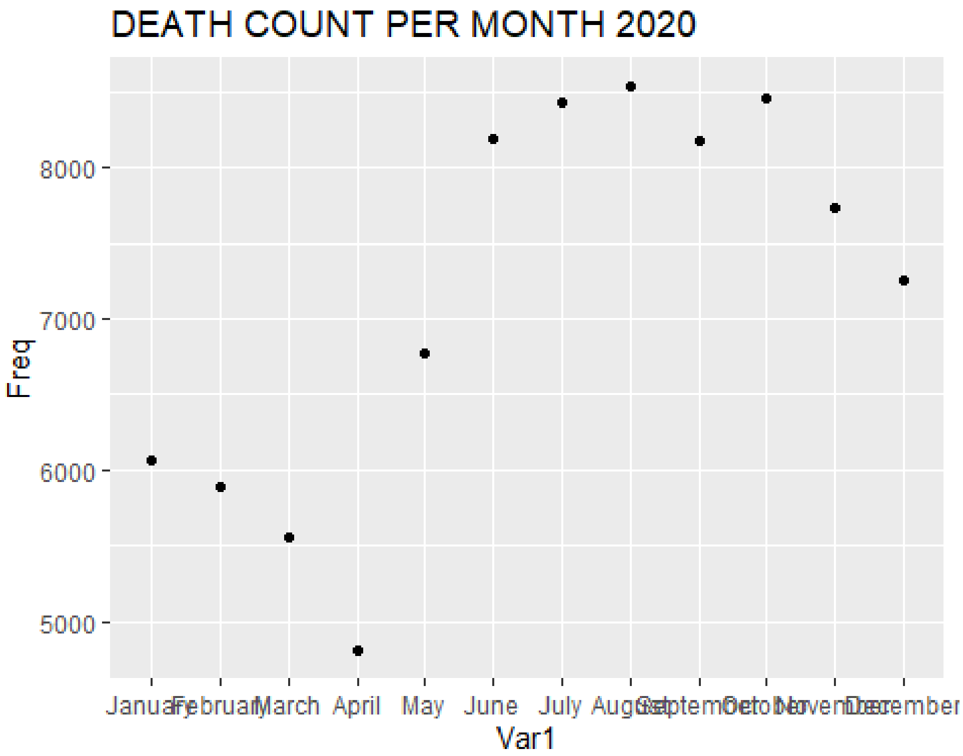
Breaking down deaths per Month:

```
n19=table(dat19$MONTHNAME)
df19=data.frame(n19)

n20=table(dat20$MONTHNAME)
df20=data.frame(n20)

sort_by_month<-factor(df19, levels = month.name)

df_month_19=data.frame(sort_by_month)
df_month_20=data.frame(sort_by_month)

ggplot(data = df19, aes(x=Var1, y=Freq))+
  geom_point(stat = "identity")+geom_line()+
  ggtitle("DEATH COUNT PER MONTH 2019")+
  scale_x_discrete(limits=month.name)

## geom_path: Each group consists of only one observation. Do you need to
adjust
## the group aesthetic?
```

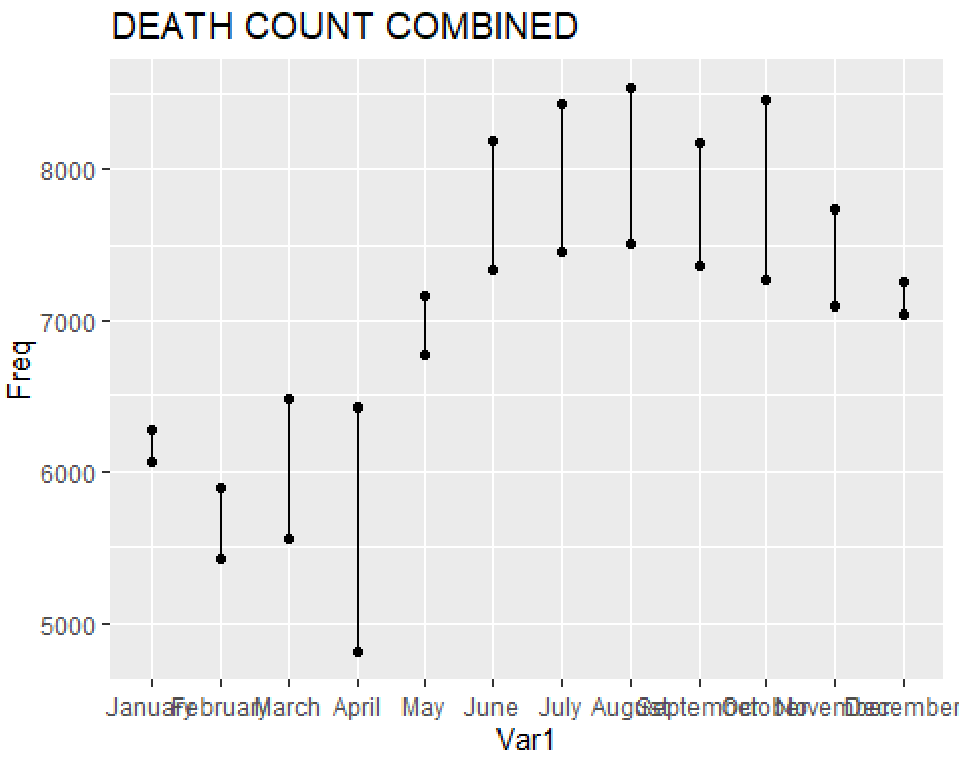


```
ggplot(data = df20, aes(x=Var1, y=Freq))+geom_point(stat = "identity")+
  geom_line()+ggtitle("DEATH COUNT PER MONTH 2020")+
  scale_x_discrete(limits=month.name)
```

```
## geom_path: Each group consists of only one observation. Do you need to
adjust
## the group aesthetic?
```



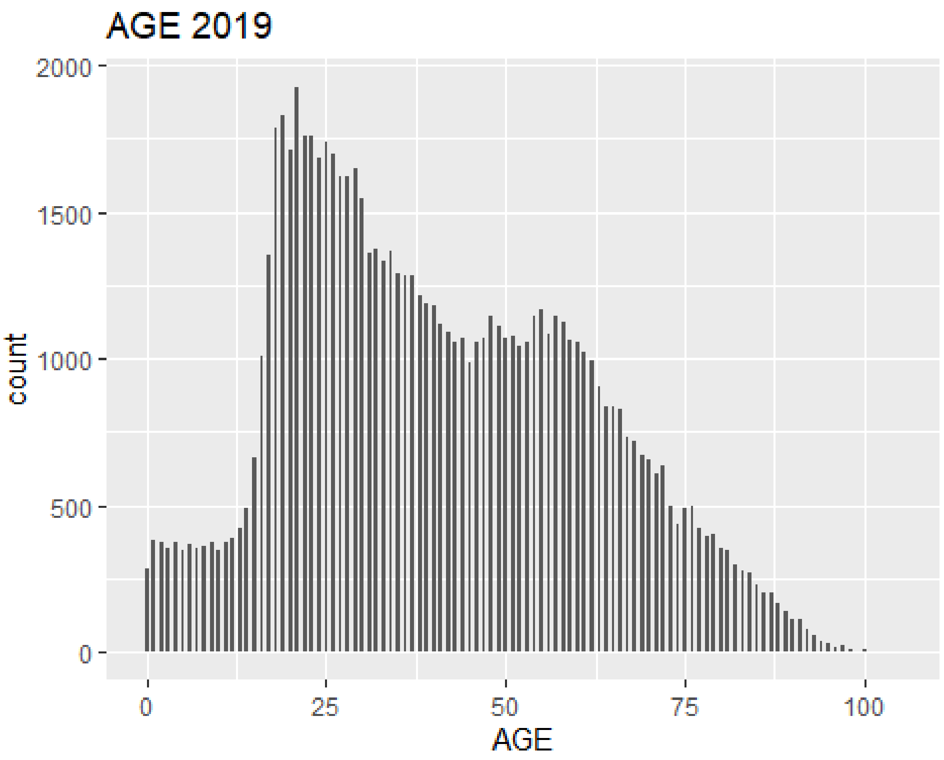
```
combined_df=rbind(df19, df20)

ggplot(data = combined_df, aes(x=Var1, y=Freq))+
  geom_point(data = df19, stat="identity")+
  geom_line()+geom_point(data = df20, stat="identity")+
  ggtitle("DEATH COUNT COMBINED")+
  scale_x_discrete(limits=month.name)
```



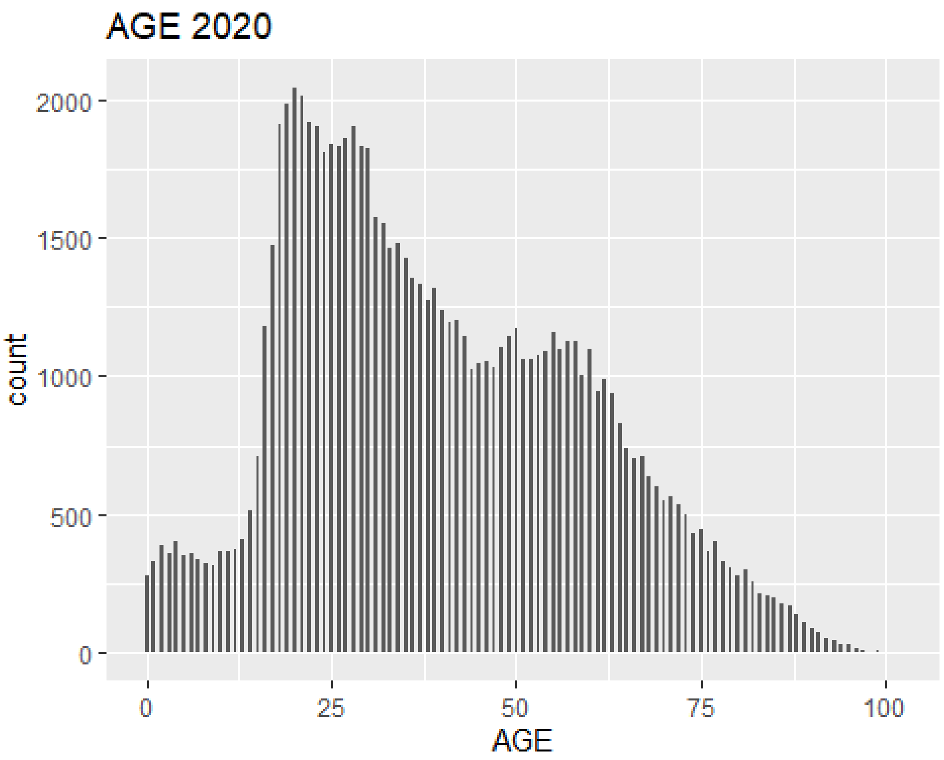
Apply filters for unknown data:

Age was inferred to be a factor in determining the differences between 2019 and 2020 car fatalities and road accidents. The graphs show the ages of drivers between the two years. # Age

```
age_19<-data.frame(AGE=dat19$AGE[dat19$AGE<115])
ggplot(data=age_19, aes(x=AGE))+geom_bar(width=.5)+ggtitle("AGE 2019")
```



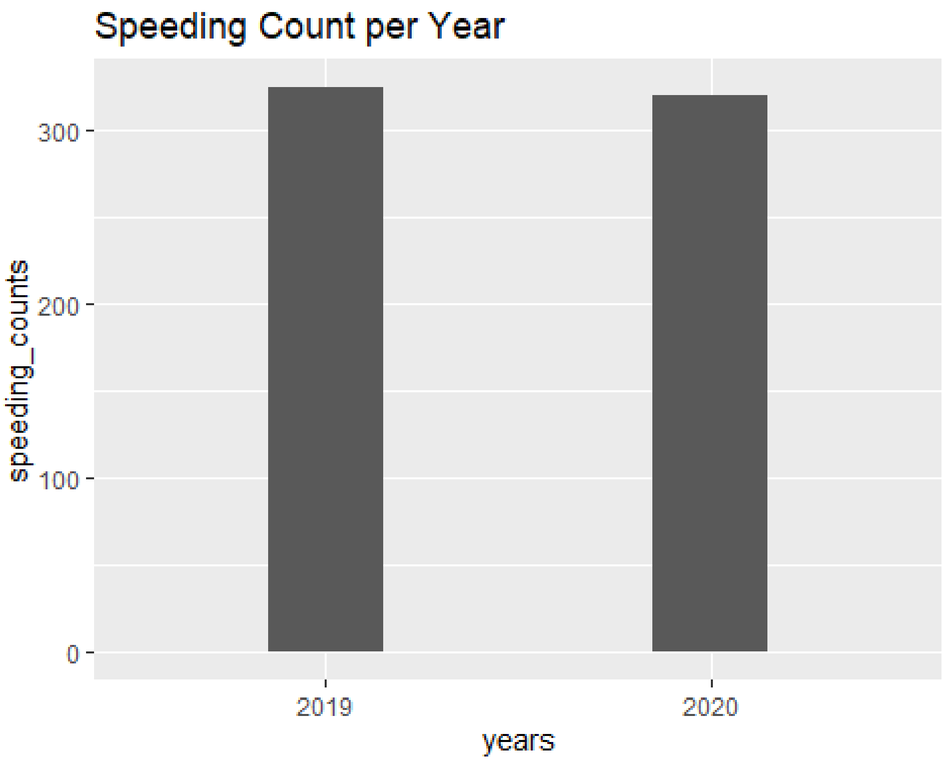
```r
age_f<-data.frame(AGE=dat20$AGE[dat20$AGE<115])
ggplot(data=age_f, aes(x=AGE))+geom_bar(width=.5)+ggtitle("AGE 2020")
```



Speeding Violations:

Speeding violations were inferred to be another factor that could impact the number of road accidents and fatalities. The graph below shows the total number of speeding violations per year and the percentage change between 2019 and 2020.

```
v2020=read.csv("./violatn2019.CSV")
v2019=read.csv("./violatn2020.CSV")



#speeding codes
viol_df2019=data.frame(table(v2019$VIOLATION))
speeding2019=viol_df2019$Freq[20]+viol_df2019$Freq[21]+viol_df2019$Freq[22]
+viol_df2019$Freq[23]+viol_df2019$Freq[24]+viol_df2019$Freq[25]

## [1] 158

viol_df2020=data.frame(table(v2020$MVIOLATN))
speeding2020=viol_df2020$Freq[20]+viol_df2020$Freq[21]+viol_df2020$Freq[22]
+viol_df2020$Freq[23]+viol_df2020$Freq[24]+viol_df2020$Freq[25]

## [1] 142

speeding_counts= c(speeding2019, speeding2020)
years = c('2019', '2020')

df=data.frame(years, speeding_counts)

ggplot(data=df, aes(x=years, y=speeding_counts))+geom_bar(stat = "identity",
width = .3)+ggtitle("Speeding Count per Year")
```



```
percent_speeding =
  ((speeding_counts[2]-speeding_counts[1])/speeding_counts[1]) * 100
percent_speeding

## [1] -1.234568
```

From the observation above, we would be hesitant to say that speeding was a major factor of why the increase in fatalities, despite having personal connections say that there has been a noticeable increase of people speeding.

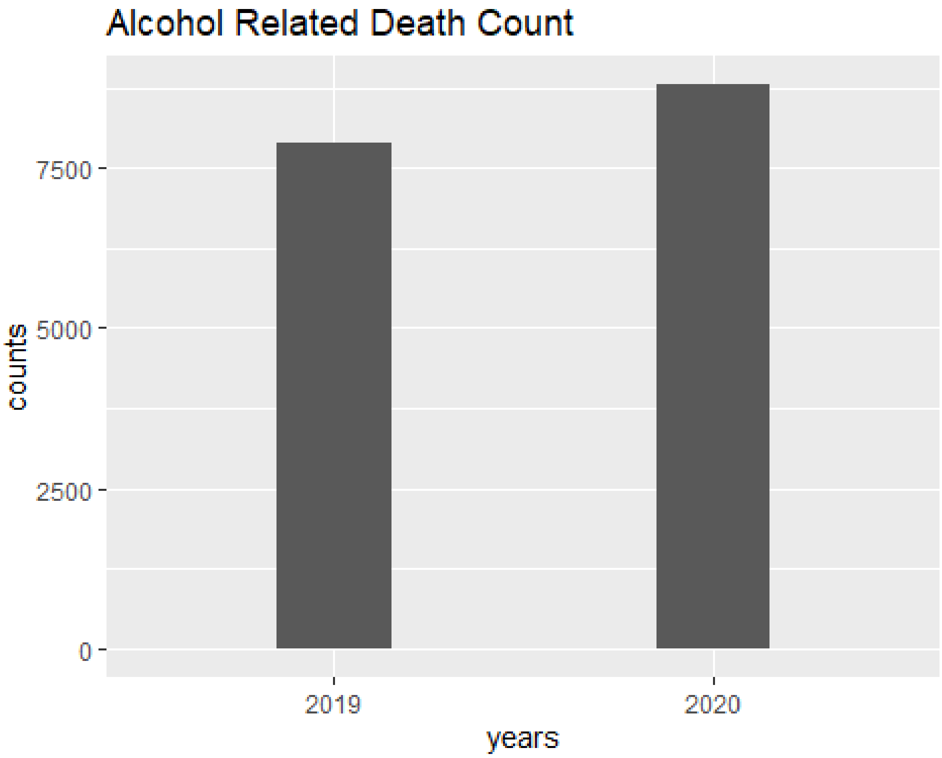
For example, here is an article relating speeding as a reason for more deaths: https://www.thelongofirm.com/posts/colorado-car-accident-fatalities-increase-2020

Accidents related to Alcohol:

Alcohol-related incidents would have been another contributing factor towards road incidents and fatalities. The graph below shows the total number of alcohol-related deaths between 2019 and 2020.

```
alc_df2019=data.frame(table(dat19$DRINKING))
alc_df2020=data.frame(table(dat20$DRINKING))

counts= c(alc_df2019$Freq[2], alc_df2020$Freq[2])
years = c('2019', '2020')

df_alc=data.frame(years, counts)
```

```
ggplot(data = df_alc, aes(x=years, y=counts))+geom_bar(stat = "identity",
width = .3)+ggtitle("Alcohol Related Death Count")
```



```
percent_alcohol = ((counts[2]-counts[1])/counts[1]) * 100
percent_alcohol

## [1] 11.62142
```

Ridge Regression Method (Math):

Ridge Regression is used to fit a model that has multicollinearity while trying to minimize the (RSS) sum of square residuals. Col-linearity in regression is the event of two or more variables being highly linearly related.

In terms of relationships between variables, our use case refers to the target or dependent variable as the number of car wrecks. The independent variable or predictor variable could be any of these; speeding, weather, age, covid cases, alcohol, drugs, etc.

*least squares regression tries to find coefficient estimates that minimize the sum of squared residuals (RSS) (https://www.statology.org/ridge-regression-in-r/)

As the tuning parameter gets larger the value of the coefficients shrink towards 0.

The Ridge Regession Esimator:

$$\sum[yi - \hat{y}_i^2] + \lambda \sum_{k=1}^{k} y_i^2$$

Another form in linear algebra.

$$\hat{\beta}(\lambda) = \left(X^T X + \lambda I_{p \times p}\right)^{-1} X^T Y$$

Linear algebra plays a significant role in understanding the lambda value as well as the rank and matrix operations. Selecting a valid lambda value is crucial to getting the best fit for the model.

$$y = \ f(x_1,\ x_2,\ x_3 \ldots \ x_n)$$

To put it simply we want to know which predictor

$$(x_i)$$

had the greatest affect on the dependent variable y.

Part 2: FARS 2020 Comparison with COVID-19

Cleaning and Reformatting Data:

The Covid-19 dataset taken from the Covid Tracking Project was reformatted to include the final positive case counts from each month between January and December 2020. Given that the original data set tracked cumulative daily counts, this was calculated by taking the total case counts on the last day of each month.

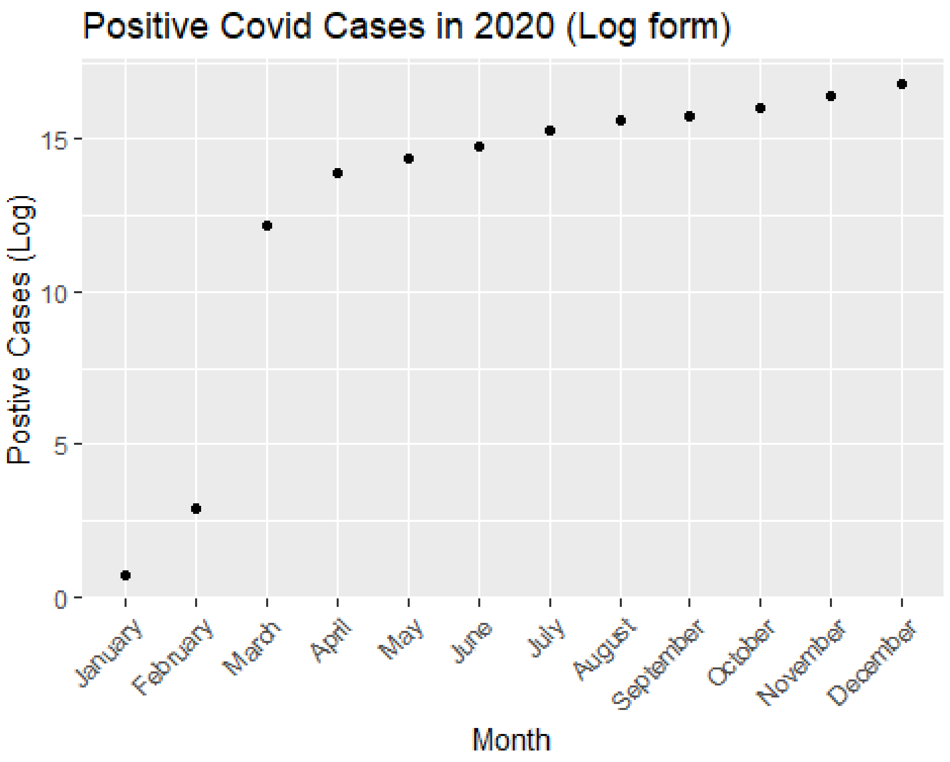
```
#This section gets the total count of covid-19 cases from January-December
2020
dat <- read.csv('national-history.csv')

#convert date to datetime
dat<-dat %>%
   mutate(date=as.Date(date))

#cut to all entries between January 2020 - December 2020
y_2020 <- filter(dat, dat$date <= '2020-12-31' )

#take the final case counts from each month
# creates an identifier with which to group
y_2020$mon_yr <- format(as.Date(y_2020$date), '%Y-%m')
#groups by created month identifier and then keeps only those rows with
last(max) date
```

```
cases <- y_2020 %>% group_by(mon_yr) %>% filter(date == max(date))
cases

## # A tibble: 12 x 18
## # Groups:   mon_yr [12]
##    date        death deathIncrease inIcuCumulative inIcuCurrently
##    <date>      <int>         <int>           <int>          <int>
##  1 2020-12-31 336802          3297           37066          23097
##  2 2020-11-30 259690          1037           30469          18807
##  3 2020-10-31 222625           958           24375           9613
##  4 2020-09-30 199080          1064           20390           6241
##  5 2020-08-31 175751           380           17537           7054
##  6 2020-07-31 145507          1324           14044          10471
##  7 2020-06-30 120258           583           10669           7419
##  8 2020-05-31 100783           654            8446           8373
##  9 2020-04-30  59646          2153            4192          13982
## 10 2020-03-31   4331           909             230           3487
## 11 2020-02-29      5             1              NA             NA
## 12 2020-01-31     NA             0              NA             NA
## # ... with 13 more variables: hospitalizedIncrease <int>,
## #   hospitalizedCurrently <int>, hospitalizedCumulative <int>, negative
<int>,
## #   negativeIncrease <int>, onVentilatorCumulative <int>,
## #   onVentilatorCurrently <int>, positive <int>, positiveIncrease <int>,
## #   states <int>, totalTestResults <int>, totalTestResultsIncrease <int>,
## #   mon_yr <chr>
```

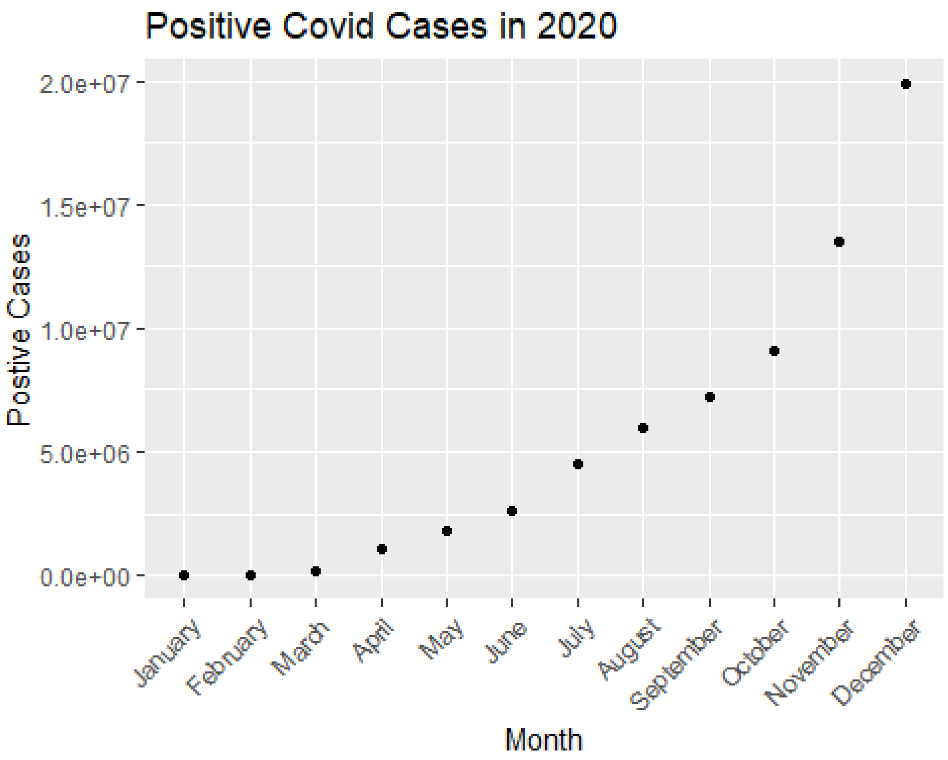
Graphing and Displaying Reformatted Data:

In graphing the number of positive covid-19 cases, a log scale was added to better show the overall number of cases nationwide.

```
#graph the positive cases

cases$logpositive = log(cases$positive)
cases$monthnames <- months(as.Date(cases$date))
cases$month<-factor(cases$monthnames, levels = month.name)

ggplot(data = cases, aes(x = month, y = logpositive)) +
  geom_point() +
  labs(x = "Month",
    y = "Postive Cases (Log)",
    title = "Positive Covid Cases in 2020 (Log form)") +
  theme(axis.text.x=element_text(angle=45,hjust=1))
```

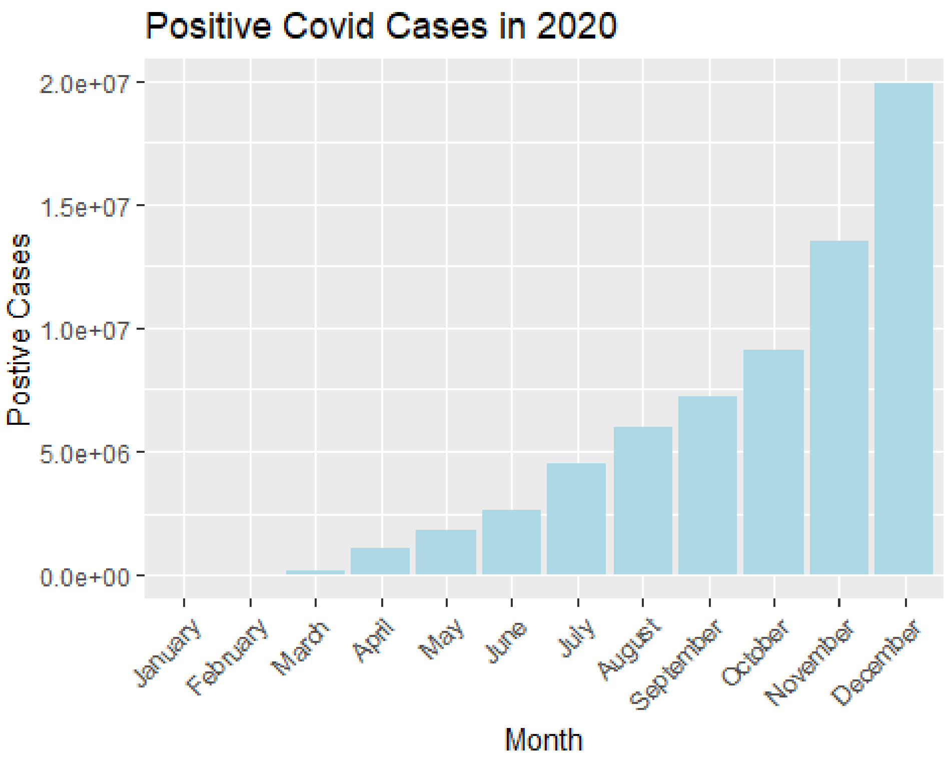


```
ggplot(data = cases, aes(x = month, y = positive)) +
  geom_point() +
  labs(x = "Month",
    y = "Postive Cases",
    title = "Positive Covid Cases in 2020") +
  theme(axis.text.x=element_text(angle=45,hjust=1))
```

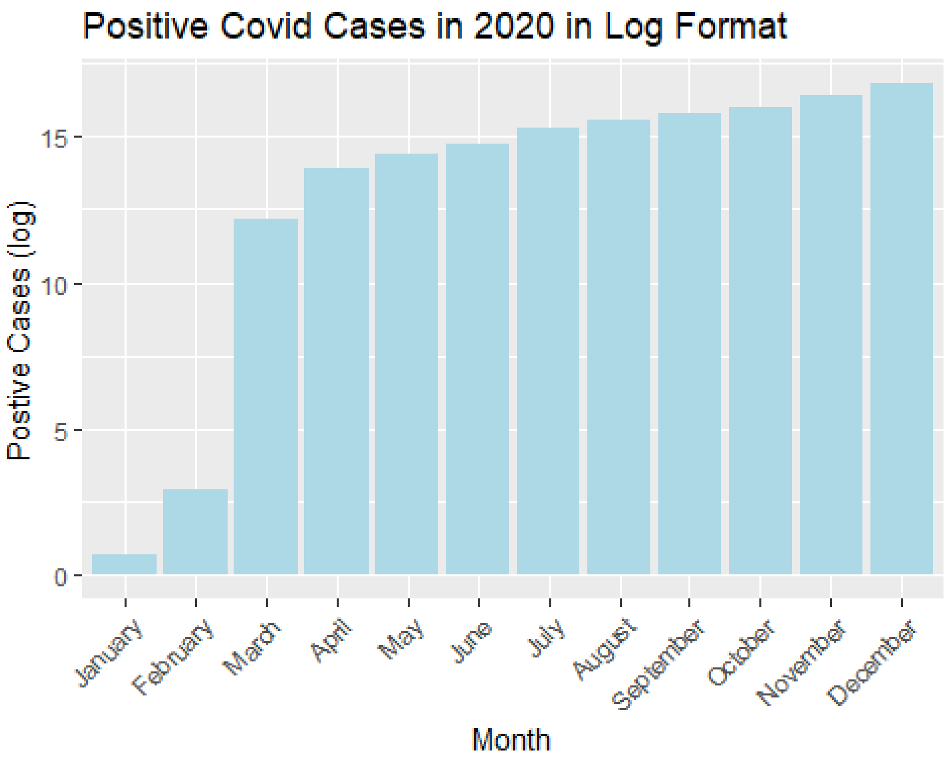


For additional comparison, a bar chart was added to reflect the total covid-19 case counts in both the normal and log scales.

```
#as a bar chart

cases$logpositive = log(cases$positive)
cases$monthnames <- months(as.Date(cases$date))
cases$month<-factor(cases$monthnames, levels = month.name)

ggplot(cases, aes(month)) + geom_bar(aes(weight=positive), position="dodge",
fill= 'lightblue') + theme(axis.text.x=element_text(angle=45,hjust=1)) +
labs(x = "Month",
    y = "Postive Cases",
    title = "Positive Covid Cases in 2020")
```



```
ggplot(cases, aes(month)) + geom_bar(aes(weight=logpositive),
position="dodge", fill= 'lightblue') +
theme(axis.text.x=element_text(angle=45,hjust=1)) +  labs(x = "Month",
    y = "Postive Cases (log)",
    title = "Positive Covid Cases in 2020 in Log Format")
```



Cleaning and Reformatting Road Incident Data:

Road accident data taken from FARS (Fatality Analysis Reporting System) 2020 was cleaned and reformatted to reflect aggregated data by month. The reformatted data was then merged with covid-19 data to show the following:

- month of the year
- total number of road incidents reported by FARS
- total positive covid-19 case counts
- the average age of individuals involved in FARS incidents
- total number of drinking violations
- total number of speeding violations

The following rmd blocks show the complete process of reformatting and merging the two data sets:

1. Read in FARS 2020 data set:

```
# total car wrecks in 2020
car_wrecks <- read.csv('person2020.csv')
```

2. Aggregate the total number of entries per month:

```
#counts of car wrecks by month
df <- data.frame(aggregate(car_wrecks, by=list(car_wrecks$MONTHNAME),
FUN=length))
names(df) <- c('Month','Crashes')

#stick it in a new data frame
columns <- c('Month','Crashes')
data2020 <- df[,columns]

data2020

##         Month Crashes
## 1       April    4811
## 2      August    8535
## 3    December    7255
## 4    February    5891
## 5     January    6071
## 6        July    8425
## 7        June    8185
## 8       March    5562
## 9         May    6778
## 10   November    7736
## 11    October    8459
## 12 September    8177
```

3. Merge (inner join) reformatted covid-19 data with FARS entry counts on month

Log scales were added for both covid-19 cases and crash data for scalable comparison.

```
#combine covid 2020 with car crashes 2020
covid <- data.frame(cases$monthnames, cases$positive)
names(covid) <- c("Month", "Cases")

total <- merge(data2020, covid, by.x = "Month", by.y="Month")
names(total) <- c("Month", "Crashes", "Cases") #naming them so we don't throw
an error

total$logCases <- log(total$Cases)
total$logCrashes<-log(total$Crashes)
total

##         Month Crashes    Cases   logCases logCrashes
## 1       April    4811  1073244 13.8861964   8.478660
## 2      August    8535  5980439 15.6040045   9.051931
## 3    December    7255 19864374 16.8044384   8.889446
## 4    February    5891       18  2.8903718   8.681181
## 5    January    6071         2  0.6931472   8.711279
## 6       July    8425   4523226 15.3247360   9.038959
## 7       June    8185   2623046 14.7798468   9.010058
## 8      March    5562    196965 12.1907813   8.623713
## 9        May    6778   1791449 14.3985353   8.821437
## 10  November    7736 13541108 16.4212407   8.953640
## 11   October    8459   9065117 16.0199443   9.042986
## 12 September    8177   7173102 15.7858488   9.009081
```

4. Age was taken from the original FARS 2020 data set and aggregated by calculating the average age per month of individuals involved in each FARS incident. Entries reflecting unknown ages '999' were dropped.

```
#Since our data set is split by month, we can get
#average age each month

car_wrecks.age<-filter(car_wrecks, AGE != 999)
#car_wrecks.age$AGE

car_wrecks.age <- car_wrecks.age %>%
  group_by(MONTHNAME) %>%
  summarise(result = mean(AGE) )

names(car_wrecks.age) <- c("Month", "Avg_Age")

car_wrecks.age
```

```
## # A tibble: 12 x 2
##    Month     Avg_Age
##    <chr>       <dbl>
##  1 April        53.7
##  2 August       50.6
##  3 December     54.6
##  4 February     48.0
##  5 January      51.9
##  6 July         52.5
##  7 June         45.9
##  8 March        52.4
##  9 May          49.2
## 10 November     58.5
## 11 October      53.4
## 12 September    52.8
```

5. Cleaned and reformatted age data was merged.

```
#join the age table on total
total.a <- merge(total, car_wrecks.age, by.x = "Month", by.y="Month")
total.a

##        Month Crashes    Cases   logCases logCrashes  Avg_Age
## 1      April    4811  1073244 13.8861964   8.478660 53.74868
## 2     August    8535  5980439 15.6040045   9.051931 50.62693
## 3   December    7255 19864374 16.8044384   8.889446 54.57355
## 4   February    5891       18  2.8903718   8.681181 48.01624
## 5    January    6071        2  0.6931472   8.711279 51.89990
## 6       July    8425  4523226 15.3247360   9.038959 52.47881
## 7       June    8185  2623046 14.7798468   9.010058 45.86492
## 8      March    5562   196965 12.1907813   8.623713 52.42142
## 9        May    6778  1791449 14.3985353   8.821437 49.21146
## 10  November    7736 13541108 16.4212407   8.953640 58.49599
## 11   October    8459  9065117 16.0199443   9.042986 53.40945
## 12 September    8177  7173102 15.7858488   9.009081 52.79774
```

#6. Drinking reports were calculated based on known reported drinking incidents, where the DRINKING column in the original FARS data set = 1. A sum for each month was calculated and inserted into a dataframe.

```
#here we aggregate Drinking reports
car_wrecks.drink<-filter(car_wrecks, DRINKING==1)
#car_wrecks.drink

car_wrecks.drink <- car_wrecks.drink %>%
  group_by(MONTHNAME) %>%
  summarise(result = sum(DRINKING) )

names(car_wrecks.drink) <- c("Month", "drink_counts")

car_wrecks.drink

## # A tibble: 12 x 2
##    Month     drink_counts
##    <chr>            <int>
##  1 April              468
##  2 August             962
##  3 December           689
##  4 February           606
##  5 January            615
##  6 July               837
##  7 June               897
##  8 March              581
##  9 May                729
## 10 November           782
## 11 October            798
## 12 September          834
```

7. Drinking counts were merged into the data frame.

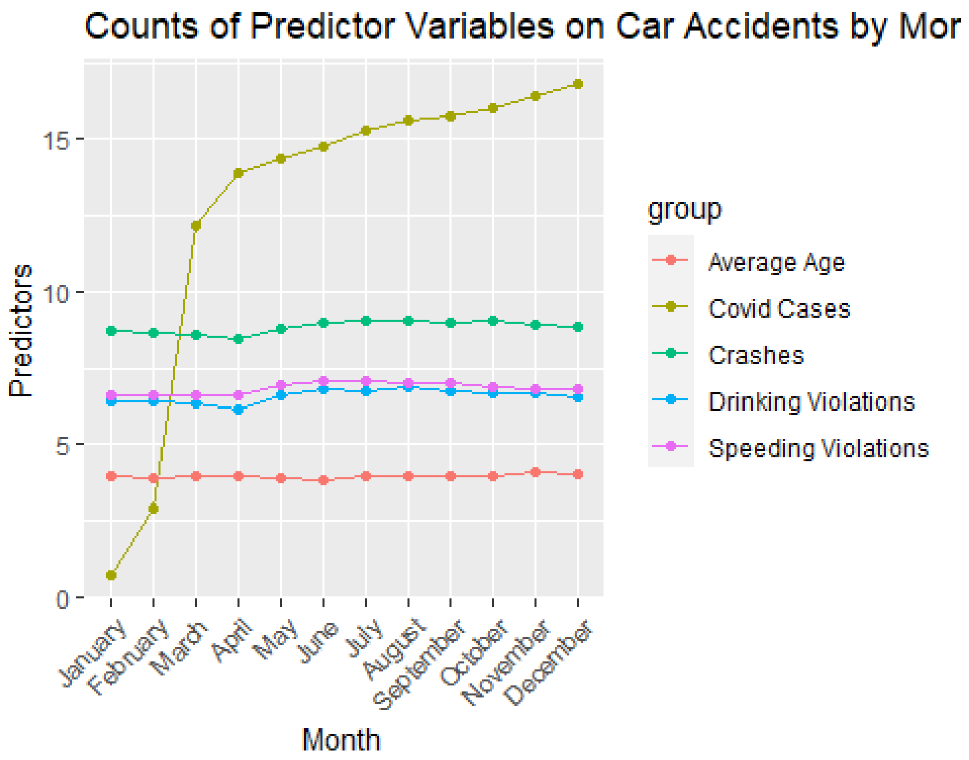
```
#merge drinking with total

total.b <- merge(total.a, car_wrecks.drink, by.x = "Month", by.y="Month")
total.b

##        Month Crashes    Cases   logCases logCrashes  Avg_Age drink_counts
## 1      April    4811  1073244 13.8861964   8.478660 53.74868          468
## 2     August    8535  5980439 15.6040045   9.051931 50.62693          962
## 3   December    7255 19864374 16.8044384   8.889446 54.57355          689
## 4   February    5891       18  2.8903718   8.681181 48.01624          606
## 5    January    6071        2  0.6931472   8.711279 51.89990          615
## 6       July    8425  4523226 15.3247360   9.038959 52.47881          837
## 7       June    8185  2623046 14.7798468   9.010058 45.86492          897
## 8      March    5562   196965 12.1907813   8.623713 52.42142          581
## 9        May    6778  1791449 14.3985353   8.821437 49.21146          729
## 10  November    7736 13541108 16.4212407   8.953640 58.49599          782
## 11   October    8459  9065117 16.0199443   9.042986 53.40945          798
## 12 September    8177  7173102 15.7858488   9.009081 52.79774          834
```

#8. Speeding related incidents were manually taken from the FARS annual report. The following vector reflects the total speeding violation counts per month in 2020. A month vector was added and speeding related incidents transformed into a data frame. Dataframe total.c reflects the final data frame transformation with all our predictor variables.

```
#add speeding-related incidents
#manually taken from page 19
#Table 4. Monthly Traffic Fatalities, by Speeding Involvement,
#Alcohol-Impaired Driving, and Passenger Vehicle Occupant Restraint Use, 2019
and 2020

speeding <- c(733, 718, 735, 761, 1007, 1189, 1151, 1089, 1079, 999, 883,
914)
month<-month.name

car_wrecks.speeding <- data.frame(month, speeding)
names(car_wrecks.speeding)<-c("Month", "Speeding")

#merge into the main df

total.c <- merge(total.b, car_wrecks.speeding, by.x = "Month", by.y="Month")
total.c

##        Month Crashes    Cases   logCases logCrashes  Avg_Age drink_counts
## 1      April    4811  1073244 13.8861964   8.478660 53.74868          468
## 2     August    8535  5980439 15.6040045   9.051931 50.62693          962
## 3   December    7255 19864374 16.8044384   8.889446 54.57355          689
## 4   February    5891       18  2.8903718   8.681181 48.01624          606
## 5    January    6071        2  0.6931472   8.711279 51.89990          615
## 6       July    8425  4523226 15.3247360   9.038959 52.47881          837
## 7       June    8185  2623046 14.7798468   9.010058 45.86492          897
## 8      March    5562   196965 12.1907813   8.623713 52.42142          581
## 9        May    6778  1791449 14.3985353   8.821437 49.21146          729
## 10  November    7736 13541108 16.4212407   8.953640 58.49599          782
## 11   October    8459  9065117 16.0199443   9.042986 53.40945          798
## 12 September    8177  7173102 15.7858488   9.009081 52.79774          834
##    Speeding
## 1       761
## 2      1089
## 3       914
## 4       718
## 5       733
## 6      1151
## 7      1189
## 8       735
## 9      1007
## 10      883
## 11      999
## 12     1079
```

The following graph shows the final monthly aggregates for each predictor in our data set.

```r
#graph the totals
total.c$logAge <- log(total.c$Avg_Age)
total.c$logDrinks <- log(total.c$drink_counts)
total.c$logSpeed <- log(total.c$Speeding)

# Reshape data frame
df_reshaped <- data.frame(x = total.c$Month,
                       y = c(total.c$logCases, total.c$logCrashes,
                            total.c$logAge, total.c$logDrinks,
total.c$logSpeed),
                       group = c(rep("Covid Cases", nrow(df)),
                                rep("Crashes", nrow(df)),
                                rep("Average Age", nrow(df)),
                                rep("Drinking Violations", nrow(df)),
                                rep("Speeding Violations", nrow(df))
                                ))

df_reshaped$x <- factor(df_reshaped$x, levels = month.name)

ggplot(df_reshaped, aes(x, y, col = group)) +  geom_line(aes(group = group))
+
  geom_point(aes(group = group)) +
  theme(axis.text.x=element_text(angle=45,hjust=1)) +  labs(x = "Month",
    y = "Predictors",
    title = "Counts of Predictor Variables on Car Accidents by Month in 2020
(Log Scale)")
```

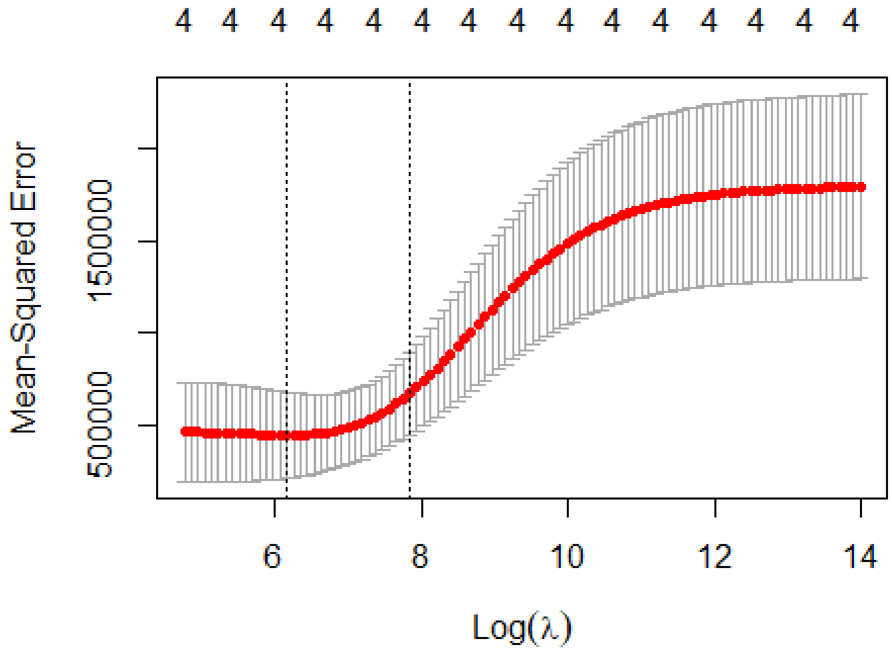


Ridge Regression Analysis:

The ridge regression was performed using the final transformed data frame "total.c". In this analysis, the response variable is defined as the total number of incidents per month reported by FARS. Columns 'Cases', 'Avg_Age', 'drink_counts', and 'Speeding' were formatted into an 'x' matrix. Using the glmnet package, a ridge regression model was fitted onto the x and y variables, and an optimal lambda value minimizing the mean squared error was chosen.

```
#begin ridge regression

#define response variable and predictors
y <- total.c$Crashes
x<-data.matrix(total.c[,c('Cases','Avg_Age','drink_counts', 'Speeding')])

#fit ridge regression model
model <- glmnet(x, y, alpha = 0)
#summary(model)

#perform k-fold cross-validation to find optimal lambda value
cv_model <- cv.glmnet(x, y, alpha = 0)

## Warning: Option grouped=FALSE enforced in cv.glmnet, since < 3
observations per
## fold
```

```
#find optimal lambda value that minimizes test MSE
best_lambda <- cv_model$lambda.min
best_lambda

## [1] 479.2966

#produce plot of test MSE by lambda value
plot(cv_model)
```

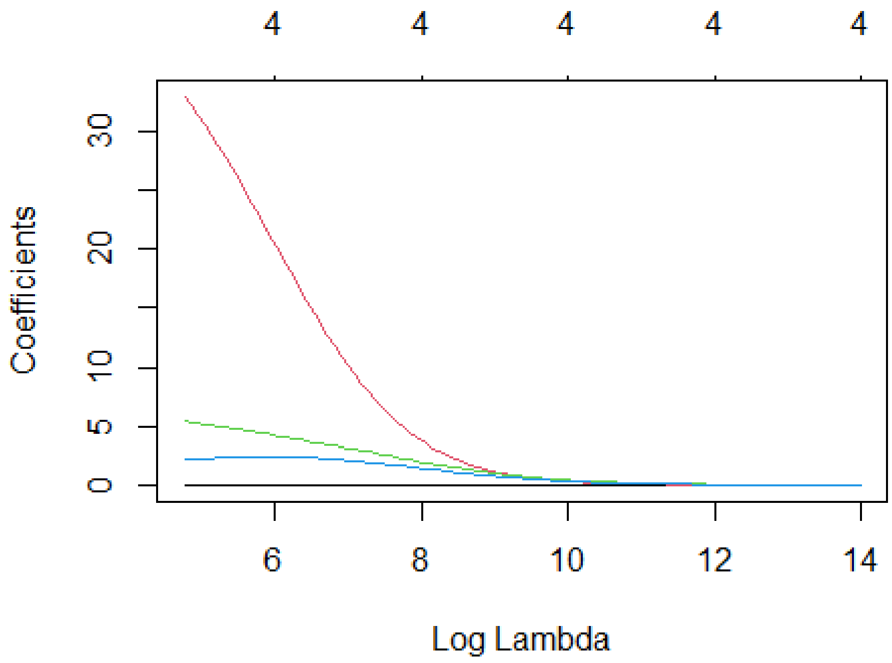


Based on the values of the coefficients of the best model, we can conclude that the number of positive covod-19 cases had very little to no effect on the overall number of FARS driving incidents in 2020. In fact it appears that the average age of drivers had the largest impact, followed by overall alcohol consumption and speeding incidents.

```
#coefficients of the best model
best_model <- glmnet(x, y, alpha = 0, lambda = best_lambda)
coef(best_model)

## 5 x 1 sparse Matrix of class "dgCMatrix"
##                         s0
## (Intercept)  7.758772e+02
## Cases        3.563755e-05
## Avg_Age      1.858331e+01
## drink_counts 4.065878e+00
## Speeding     2.386681e+00
```

The model below shows the Ridge trace plot for each of the coefficients as the log lambda increases. As lambda increases, the coefficients trend towards 0.

```
#produce Ridge trace plot
plot(model, xvar = "lambda")
```



Verifying our Model

To verify the accuracy of our model, we predicted some y-values using our best model and used these values to calculate the total sum of squares (sst) and the sum of squared errors (sse). From these values, we know that the R^2 value of our model is about 0.94. Thus, our model explains about 94% of the variation in our data.

```
#use fitted best model to make predictions
y_predicted <- predict(model, s = best_lambda, newx = x)

#find SST and SSE
sst <- sum((y - mean(y))^2)
sse <- sum((y_predicted - y)^2)

#find R-Squared
rsq <- 1 - sse/sst
rsq

## [1] 0.9177653
```

Because the original data sets had to be aggregated in order to run the model, there may have been some loss in the data when translating into the model. Many entries in the FARS data set had unknown values. Thus there may have been insufficient data for the Drinking, Age, and Speeding columns, and the coefficients for the best-fit Ridge regression model may not be representative of the true population values.

Finally, there may have been many other predictors not considered within this report that could have affected our conclusion. Nevertheless, the empirical evidence of the study seems to suggest that there were differences in the number of vehicle accidents and fatalities between 2019 and 2020, and that the COVID-19 pandemic did not appear to have a significant impact on the number of road incidents in 2020. The number of overall car wreck incidents went down from 2019-2020 however, the total number of car wreck fatalities increased.[2]

Citations

[0] Ridge Regression in R (Step-by-Step) *https://www.statology.org/ridge-regression-in-r/*
[1] Elements of Statistical Learning Data Mining, Inference, and Prediction

[2] *https://www.nhtsa.gov/press-releases/2020-traffic-crash-data-fatalities*

[3] COVID dataset https://covidtracking.com/data/national

[4] FARS https://www.nhtsa.gov/file-downloads?p=nhtsa/downloads/FARS/