Project- Motor Vehicle Collisions - Crashes

By
Nithin Reddy.P: 002896440

Subject: ALY6015-Intermediate Analytics
Under the guidance of Dr. Harpreet Sharma
Submission Date:17-Feb-2024

Table of Contents

Introduction:

The dataset comprises information on motor vehicle collisions in New York City, collected by the NYPD. It includes details such as crash date, time, borough, location, injuries, fatalities, contributing factors, and vehicle types. The dataset aims to support traffic safety analysis and initiatives like Vision Zero, tracking incidents to enhance understanding and address road safety concerns.

Data Cleaning:

The dataset is very comprehensive and contains several null and empty values for each observation [Fig-1], we have used clean names to have uniformity and identified the we need to clean. We have assigned NA values to empty cells and removed them accordingly along with unwanted columns.

Further [Fig-2]; we have created a new column for fatalities for extensive analysis and also converted date and year from our dataset to a unique column for further analysis.

Exploratory Data Analysis:

EDA plays a pivotal role in gaining insights about the classification of the data and also helps us to understand the trends and insights of available data. The main aim is to find patterns that help us to address the questions which we intend to explore further in this project.

We have used summary and describe function understand the distribution of the data [Fig-3]. To understand the relation between observation, we have built a correlation matrix by creating a subset which contains only numeric values. We then plotted it using correlation plot [Fig-4].

From the correlation plot we can observe the relation between variables. 1 is the positively strong relation whereas -1 is negatively strong. From the fig we can say that persons injured has strong relation with motorists injured and week relation with persons killed, similarly fatalities have strong relation with persons killed as week relation with motorists injured.

We have also build a subset model to find best model for regression, from the results we came to conclusion that the best regression model is with number_of_pedestrians_injured, number_of_motorist_injured, number_of_persons_killed, and number_of_cyclist_injured [Fig-9].

To understand the year wise distribution of crashes we have built a bar graph, Fig-5 and realized that 2021 had most crashes followed by 2022 and 2023. We tried to understands no of deaths in different categories over the years [Fig-6]. We observed that the most of the deaths were of motorists from 2021-2023.

Several factors would result in a crash, we tried to identify top 10 factors using bar plot [Fig-7], we observed that most of the crashes are caused due to distraction, followed by right-off way, unspecified reasons, unsafe speed, following closely and lane usage. Finally, we created a bar plot [Fig-8], to understand which Borough has more Fatalities. From the graph we could observe that there many fatalities in Brooklyn and Qeens, followed by Bronx and Manhattan.

Preliminary Analysis:

Multiple Linear Regression:

We are trying to understand what factors contribute the most to the number of injuries in motor vehicle collisions?

From Fig-4, we could see relation between several variables, by conducting multiple linear regression we would be able to predict the contributing factors which could cause injuries in the future.

Please refer to Fig-9, we built a subset model to identify the best sub set of 4 predictor variables and identified number_of_pedestrians_injured,number_of_motorist_injured,number_of_persons_killed,and number_of_cyclist_injured, would create a best model.

Based on the results, we created a regression model [Fig-10]. The coefficient of number of people injured is 0.988, when other variables are constant, which mean for every additional pedestrian injured we expect no of persons injured to increase by 0.988. Similarly, for cyclist injured and motorist injured the coefficients are 0.96 and 0.98. The coefficient of persons killed is -0.022, however the p-value is greater than 0.05, which means is statistically insignificant and we do not have evidence to conclude the linear relation with persons injured.

The Multiple R-squared suggest that the 95.3% of the variance in no of persons injured can be explained the predictors in the model. While, the small F-statistic and p-value suggests that the model is statistically significant.

Regression Diagnostics:

Regression Plots:

Residuals Vs Fitted Plot: This plot is used to check linearity. From the plot we can observe that there is positive relation, but the pattern is not clear which suggest linearity [Fig-11].

Q-Q Plot: This plot is used to check normality with residuals which is error. From the plot we can observe that there is normality but some deviation among the tails [Fig-12].

Scale-Location Plot: This plot shows whether the residuals are equally speared within the range. From the plot we can say that homoscedasticity exists but after a certain point in x axis the residuals began to spread wider [Fig-13].

Residuals Vs Leverage: This plot helps us to find the outliers. Here from the plot we have few points which are on the cook’s distance, which means that these residuals might impact the R-squared. In this case despite removing the outlies we do not have any significant impact on the regression model [Fig-14].

Multi-collinearity: Multi-collinearity is used to check if we have any correlation between the independent variables. If the VIF is greater than 2 it indicates that the correlation exists [Fig-15]; the VIS values of all independent variables are below the threshold supporting that there is no multicollinearity in the model.

Assessing Outliers: It is essential to find the outliers and remove them to build an accurate model. I have used the outliers function to find them and consequently removed them. The Residuals Vs Leverage plot have shown the same outliers and they have been treated [Fig-16].

Model-2:

Since the outliers are removed, we are re-assessing to find its efficiency. Form the results [Fig-17], we could see that there is no difference in both models. We can also see the same in while comparing using AIC, they both are the same, [Fig-18].

ANOVA (Analysis of Variance):

Our main aim is to Predicting the severity of injuries in vehicle crashes.

Hypothesis:

H0: There is no significant difference in the mean number of persons killed among different levels of contributing_factor_vehicle

H1: At least one group's mean severity is different from the others.

From Fig-19, we can clearly say that the p-value of contributing_factor_vehicle_1 is very small, which suggests that there is strong evidence against null hypothesis and conclude that there is significant difference in means. On the other hand, contributing_factor_vehicle_2 is higher than the significance level, which suggests that there is no significant difference in means.

The ANOVA model suggests that contributing_factor_vehicle_1 has significant effect on mean number of persons killed, while contributing_factor_vehicle_2 does not.

Chi-Square Test for Independence

Our main aim is to check to conduct spatial analysis of crashes in different boroughs.

Hypothesis:

H0: There is no association between the borough and the occurrence of fatalities in traffic collisions.

H1: There is an association between the borough and the occurrence of fatalities in traffic collisions.

From Fig-20, we could clearly observe that the p-value is really smally which suggest that there is strong evidence against null hypothesis and conclude that there is significant association between the borough and the occurrence of fatalities in traffic collisions.

The Chi-Square model suggests that occurrence of fatalities in traffic collisions is not evenly distributed across different boroughs and that there is a significant relationship between the borough and the likelihood of fatalities occurring in traffic collisions.

Predictive Analysis:

Here we are looking to identify What are the key contributing factors to fatalities in motor vehicle collisions, and how do the models differ in their selection of predictors? Furthermore, through the utilization of lasso and ridge regression models,

GLM Regression Report

Introduction:

Traffic accidents are a significant public safety concern, with fatal outcomes posing a grave risk to individuals and communities. Understanding the factors contributing to fatalities in vehicular incidents is crucial for developing effective preventive measures and enhancing road safety. In this analysis, a Generalized Linear Model (GLM) regression was employed to investigate the relationship between various contributing factors and the likelihood of fatalities.

Data Preparation:

Before constructing the GLM regression model, the dataset was split into training and testing sets to facilitate model validation. Categorical variables, specifically **contributing_factor_vehicle_2**, were converted into factors to enhance the model's interpretability and predictive performance.

Model Building:

The GLM logistic regression model was constructed using the **glm** function. The binary outcome variable was defined as the presence or absence of fatalities (**fatalities ~ contributing_factor_vehicle_1 + contributing_factor_vehicle_2**). The logit link function was chosen to model the relationship between the predictors and the binary response variable.

Model Summary:

The model summary provides detailed information about the coefficients, standard errors, z-values, and p-values associated with each contributing factor. This information is crucial for understanding the strength and significance of the relationships between these factors and the likelihood of fatalities [Fig-21].

Confusion Matrix Analysis Summary:

The model exhibits an accuracy of 65.65%, with a Kappa value of 0.1993, indicating fair agreement beyond random chance. Sensitivity (32.00%) suggests room for improvement in correctly identifying positive instances[Fig-22]. The model demonstrates a significant difference from random chance (McNemar's Test P-Value < 2.2e-16). Balancing sensitivity and specificity, the balanced accuracy is 59.08%. Further exploration of model parameters and features is recommended to enhance predictive performance, particularly in capturing positive instances.

Analysis on Metrics:

Recall:

Recall for the testing set is approximately 0.5840, indicating that the model correctly identifies about 58.40% of the actual fatalities [Fig-23].

Precision:

Precision for the testing set is approximately 0.3199754, indicating that when the model predicts a fatality, it is correct about 31.99% of the time [Fig-24].

F1 Score:

The F1 Score, which balances precision and recall, is approximately 0.4136767 for both the training and testing sets [Fig-25].

These metrics collectively provide insights into the model's performance, emphasizing the trade-off between correctly identifying fatalities and avoiding false positives. Further optimization and tuning may be considered to enhance the model's predictive capabilities.

AUC Calculation:

The AUC value is approximately 0.6465. This suggests that the model has a moderate discriminatory ability, with a higher probability of correctly ranking a randomly chosen positive instance higher than a randomly chosen negative instance. The AUC is a quantitative measure of the overall performance of the model[Fig-26].

ROC Curve:

We also plot a Receiver Operating Characteristic (ROC) curve for a model that predicts traffic fatalities. It shows the trade-off between the true positive rate (TPR) and the false positive rate (FPR) at different classification thresholds.

The blue line in the graph represents the ROC curve for your model. The closer the curve is to the top left corner, the better the model's performance. In this case, the curve is close to the top left corner, which indicates that the model has good performance.

Overall, the ROC curve suggests that our model is a good predictor of traffic fatalities. It has a high AUC and is able to correctly identify a high proportion of true positives without generating a high proportion of false positives[Fig-27].

Ridge and Lasso Regression Analysis

In this section, we delve into the application of Ridge and Lasso regression on our dataset related to fatalities. The analysis focuses on a subset of the dataset, including contributing factors, vehicle types, and the target variable, "fatalities." The dataset is split into training and testing sets, with a 70-30 split, and a random seed is set to 123 for reproducibility. Due to the dataset's large size, only the first 1000 rows are considered for both training and testing [Fig-28].

Data Preparation and Subset Selection

The process involves transforming the data into matrices, a prerequisite for the glmnet package. The model matrix is created, excluding the target variable ("fatalities"), to prepare feature matrices for both training and testing sets. The target variable is assigned to "train_y" and "test_y" for the training and testing sets, respectively.

Lasso Regression

Lasso regression is performed with cross-validation to identify optimal lambda values. Two Lasso models are built using these optimal lambda values, and their respective deviance explained and non-zero coefficients are displayed.

Ridge Regression

Additionally, a Ridge model is constructed using the optimal lambda value for Lasso at 1se. The model's performance is evaluated using Root Mean Squared Error (RMSE) on both the training and test sets.

Model Evaluation and Comparison

The Lasso model with Lambda.min yields a training RMSE of 0.427219 and a test RMSE of 0.5127028. In contrast, the Ridge model with Lambda.1se exhibits a training RMSE of 0.4104434 and a test RMSE of 0.6806565. These results suggest that, in this instance, the Ridge model performs slightly better on the test set compared to the Lasso model.

Hyperparameter Tuning and Model Fitting

To identify the optimal regularization parameter (lambda) values for Lasso regression, we employed cross-validation (cv.glmnet) with 10 folds. The random seed was set to 123 for reproducibility [Fig-29].

Lasso Regression Models

Lasso Model with Lambda.min:

- Optimal Lambda: 0.0141
- Deviance Explained: 21.96%
- Number of Non-Zero Coefficients: 49
- Root Mean Squared Error (RMSE) on Training Data: 0.427219
- RMSE on Test Data: 0.5127028[Fig-30]

Lasso Model with Lambda.1se:

- Optimal Lambda: 0.02464
- Deviance Explained: 16.43%
- Number of Non-Zero Coefficients: 21

- RMSE on Training Data: 0.4424
- RMSE on Test Data: 0.4831[Fig-31,32]

Ridge Regression Model

Ridge Model with Lambda.1se:

- Optimal Lambda: 0.02464
- Deviance Explained: 27.97%
- Number of Non-Zero Coefficients: 127
- RMSE on Training Data: 0.4104434
- RMSE on Test Data: 0.6806565[Fig-33,34]

Ridge Model with Lambda.min:

- Optimal Lambda: 0.0141
- Deviance Explained: 28.01%
- Number of Non-Zero Coefficients: 127
- RMSE on Training Data: 0.4103256
- RMSE on Test Data: 0.6881625 [Fig-33,34]

Plot:

The code fits a LASSO regression model to predict a target variable called "fatalities" using data from two other variables, "contributing_factor_vehicle_1" and "contributing_factor_vehicle_2". It then performs a 10-fold cross-validation to select the optimal value of the "lambda" parameter, which controls the amount of shrinkage applied to the model coefficients[Fig-35].

The graph is a plot of the mean squared error (MSE) for different values of lambda. Each line represents one-fold of the cross-validation. The blue line shows the average MSE across all folds. The vertical dashed lines indicate the values of lambda that minimize the mean squared error (lambda.min) and the one-standard-error rule (lambda.1se).

Summary and Interpretation

Lasso models with different lambda values were fitted, providing insights into feature selection and their impact on model performance. The Ridge model demonstrated a higher deviance explained compared to the Lasso models but resulted in slightly higher RMSE on the test set. Lasso's ability to induce sparsity might be beneficial for feature selection, while Ridge tends to provide stable coefficients.

Conclusion:

We conducted an extensive analysis to delve into the factors influencing the severity of injuries in motor vehicle collisions and to explore the spatial distribution of crashes across various boroughs. Our regression analysis revealed that approximately 95% of the variance in the number of persons injured can be explained, with statistically significant positive associations found between injury severity and factors such as the number of pedestrians, cyclists, and motorists injured. Moreover, through ANOVA, we identified that contributing factors under contributing_factor_vehicle_1 significantly impact the mean number of persons killed in collisions, emphasizing the necessity of considering these factors when assessing injury severity and devising targeted interventions. Furthermore, our spatial analysis using Chi-Square demonstrated a significant association between borough and the occurrence of fatalities in traffic collisions.

In our predictive analysis, our objective was to pinpoint the key contributing factors to fatalities in motor vehicle collisions and evaluate the performance of various models in predicting these outcomes. Employing a Generalized Linear Model (GLM) regression, we discovered that factors related to vehicles' contributing factors have a significant influence on the likelihood of fatalities. While our model displayed moderate accuracy, with an overall accuracy rate of 65.65% and a Kappa value of 0.1993 indicating fair agreement beyond random chance, there remains room for enhancement in sensitivity, as only 32.00% of positive instances were correctly identified. Additional metrics such as recall, precision, and F1 score provided further insights into the model's performance, highlighting the delicate balance between correctly identifying fatalities and avoiding false positives. The GLM regression model achieved an AUC value of approximately 0.6465, signifying moderate discriminatory ability.

Furthermore, we explored the utilization of Ridge and Lasso regression models on a subset of the dataset. Both models offered valuable insights, with the Ridge model slightly outperforming the Lasso model on the test set. Hyperparameter tuning using cross-validation enabled us to pinpoint optimal lambda values for the Lasso model, resulting in enhanced predictive performance.

Recommendations:

Based on our analysis, several recommendations can be made to improve road safety and mitigate the severity of injuries and fatalities in motor vehicle collisions

Targeted Educational Campaigns:

Develop targeted awareness campaigns focusing on significant factors such as "Backing Unsafely," "Lost Consciousness," and "Traffic Control Disregarded." Educate drivers about the risks associated with specific behaviors and encourage responsible driving practices.

Law Enforcement Prioritization:

Collaborate with law enforcement agencies to prioritize monitoring and enforcement of regulations related to statistically significant factors. Implement stricter penalties or targeted interventions for behaviors like "Passing Too Closely" or "Oversized Vehicle" violations.

Infrastructure Improvements:

Analyze collision data related to contributing factors like "Pavement Defective" and "Lane Marking Improper/Inadequate." Invest in infrastructure improvements, such as road maintenance and clear lane markings, to reduce the risk of accidents.

Medical Screening for Drivers:

Explore initiatives for regular medical screenings for drivers, especially focusing on conditions associated with factors like "Lost Consciousness" or "Illness."

Technology Integration:

Encourage the use of advanced safety technologies, such as collision avoidance systems, to mitigate the impact of contributing factors like "Following Too Closely" or "Unsafe Lane Changing."

Community Engagement:

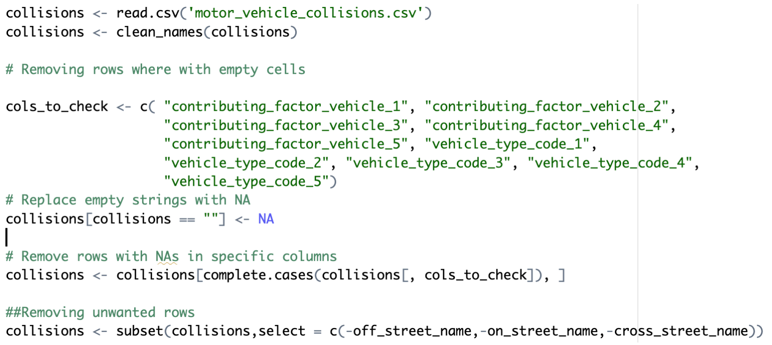
Involve the community in road safety initiatives by organizing workshops, events, or forums to discuss the importance of responsible driving and adherence to traffic rules.

Regular Monitoring and Evaluation:

Establish a system for continuous monitoring and evaluation of road safety measures. Periodically review the effectiveness of implemented interventions and adjust strategies based on evolving trends in contributing factors. By addressing these strategies, policymakers, law enforcement, and community stakeholders can work collaboratively to create a safer road environment and reduce the occurrence of motor vehicle fatalities

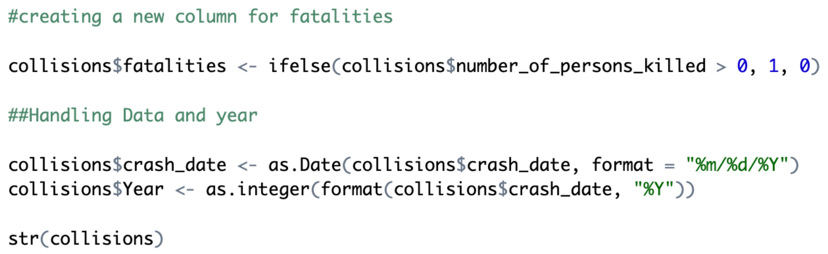
Appendix:

Fig-1: Data Cleaning



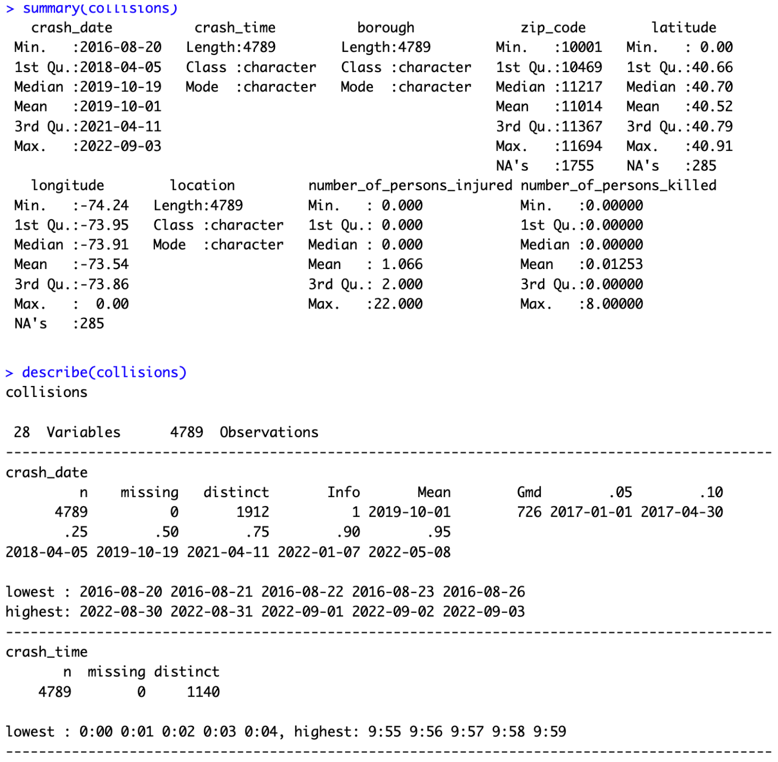
```
collisions <- read.csv('motor_vehicle_collisions.csv')
collisions <- clean_names(collisions)

# Removing rows where with empty cells

cols_to_check <- c( "contributing_factor_vehicle_1", "contributing_factor_vehicle_2",
                    "contributing_factor_vehicle_3", "contributing_factor_vehicle_4",
                    "contributing_factor_vehicle_5", "vehicle_type_code_1",
                    "vehicle_type_code_2", "vehicle_type_code_3", "vehicle_type_code_4",
                    "vehicle_type_code_5")
# Replace empty strings with NA
collisions[collisions == ""] <- NA

# Remove rows with NAs in specific columns
collisions <- collisions[complete.cases(collisions[, cols_to_check]), ]

##Removing unwanted rows
collisions <- subset(collisions,select = c(-off_street_name,-on_street_name,-cross_street_name))
```

Fig-2: Data Preparation



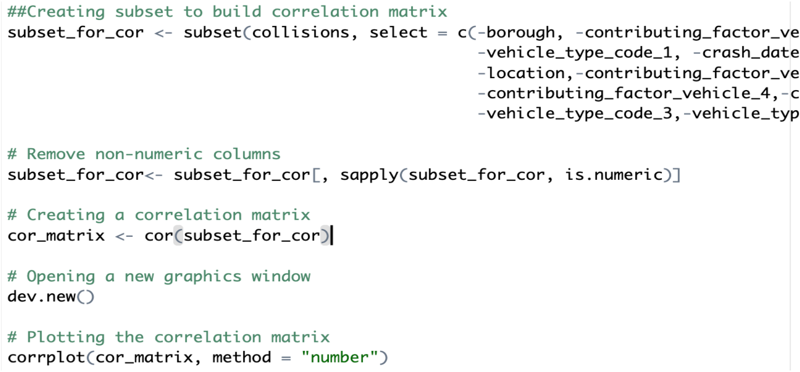
```
#creating a new column for fatalities

collisions$fatalities <- ifelse(collisions$number_of_persons_killed > 0, 1, 0)

##Handling Data and year

collisions$crash_date <- as.Date(collisions$crash_date, format = "%m/%d/%Y")
collisions$Year <- as.integer(format(collisions$crash_date, "%Y"))

str(collisions)
```

Fig-3: Observations



```
> summary(collisions)
   crash_date          crash_time          borough             zip_code        latitude    
 Min.   :2016-08-20   Length:4789        Length:4789        Min.   :10001   Min.   : 0.00  
 1st Qu.:2018-04-05   Class :character   Class :character   1st Qu.:10469   1st Qu.:40.66  
 Median :2019-10-19   Mode  :character   Mode  :character   Median :11217   Median :40.70  
 Mean   :2019-10-01                                         Mean   :11014   Mean   :40.52  
 3rd Qu.:2021-04-11                                         3rd Qu.:11367   3rd Qu.:40.79  
 Max.   :2022-09-03                                         Max.   :11694   Max.   :40.91  
                                                            NA's   :1755    NA's   :285    
   longitude        location         number_of_persons_injured number_of_persons_killed
 Min.   :-74.24   Length:4789        Min.   : 0.000            Min.   :0.00000         
 1st Qu.:-73.95   Class :character   1st Qu.: 0.000            1st Qu.:0.00000         
 Median :-73.91   Mode  :character   Median : 0.000            Median :0.00000         
 Mean   :-73.54                      Mean   : 1.066            Mean   :0.01253         
 3rd Qu.:-73.86                      3rd Qu.: 2.000            3rd Qu.:0.00000         
 Max.   :  0.00                      Max.   :22.000            Max.   :8.00000         
 NA's   :285                                                                           

> describe(collisions)
collisions 

 28  Variables      4789  Observations
---------------------------------------------------------------------------------------------
crash_date 
         n    missing   distinct       Info       Mean        Gmd        .05        .10 
      4789          0       1912          1 2019-10-01        726 2017-01-01 2017-04-30 
       .25        .50        .75        .90        .95 
2018-04-05 2019-10-19 2021-04-11 2022-01-07 2022-05-08 

lowest : 2016-08-20 2016-08-21 2016-08-22 2016-08-23 2016-08-26
highest: 2022-08-30 2022-08-31 2022-09-01 2022-09-02 2022-09-03
---------------------------------------------------------------------------------------------
crash_time 
       n  missing distinct 
    4789        0     1140 

lowest : 0:00 0:01 0:02 0:03 0:04, highest: 9:55 9:56 9:57 9:58 9:59
---------------------------------------------------------------------------------------------
```

Fig-4: Correlation Plot



```
##Creating subset to build correlation matrix
subset_for_cor <- subset(collisions, select = c(-borough, -contributing_factor_vehicle_1,
                                                -vehicle_type_code_1, -crash_date,-crash_time,-zip_code,-Year,
                                                -location,-contributing_factor_vehicle_2,-contributing_factor_vehicle_3,
                                                -contributing_factor_vehicle_4,-contributing_factor_vehicle_5,-vehicle_type_code_2,
                                                -vehicle_type_code_3,-vehicle_type_code_4,-vehicle_type_code_5,-latitude,-longitude))

# Remove non-numeric columns
subset_for_cor<- subset_for_cor[, sapply(subset_for_cor, is.numeric)]

# Creating a correlation matrix
cor_matrix <- cor(subset_for_cor)

# Opening a new graphics window
dev.new()

# Plotting the correlation matrix
corrplot(cor_matrix, method = "number")
```

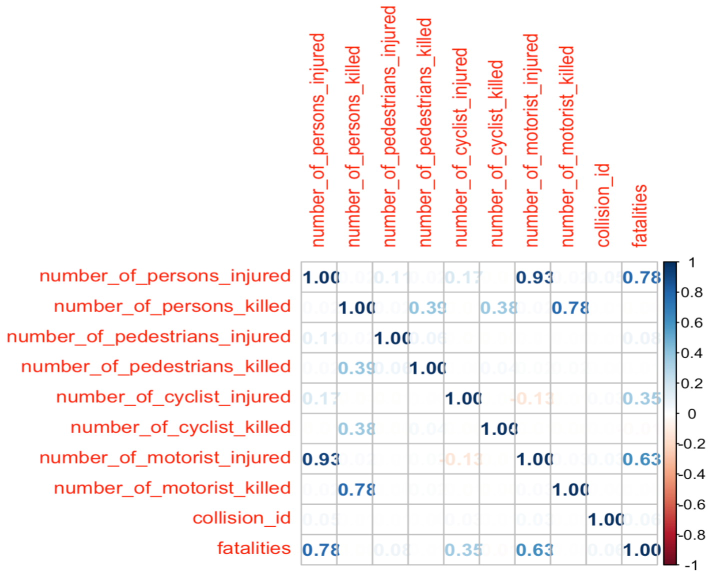
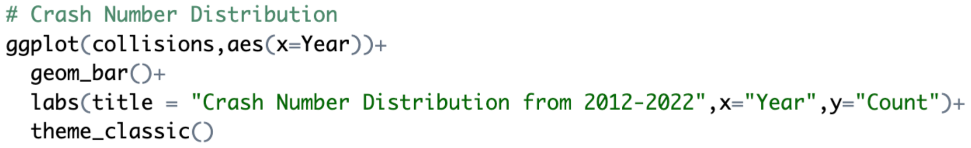


Fig-5: Crash Distribution



```
# Crash Number Distribution
ggplot(collisions,aes(x=Year))+
  geom_bar()+
  labs(title = "Crash Number Distribution from 2012-2022",x="Year",y="Count")+
  theme_classic()
```

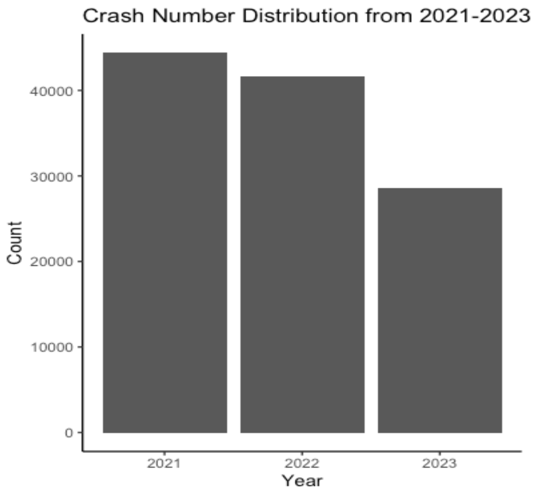
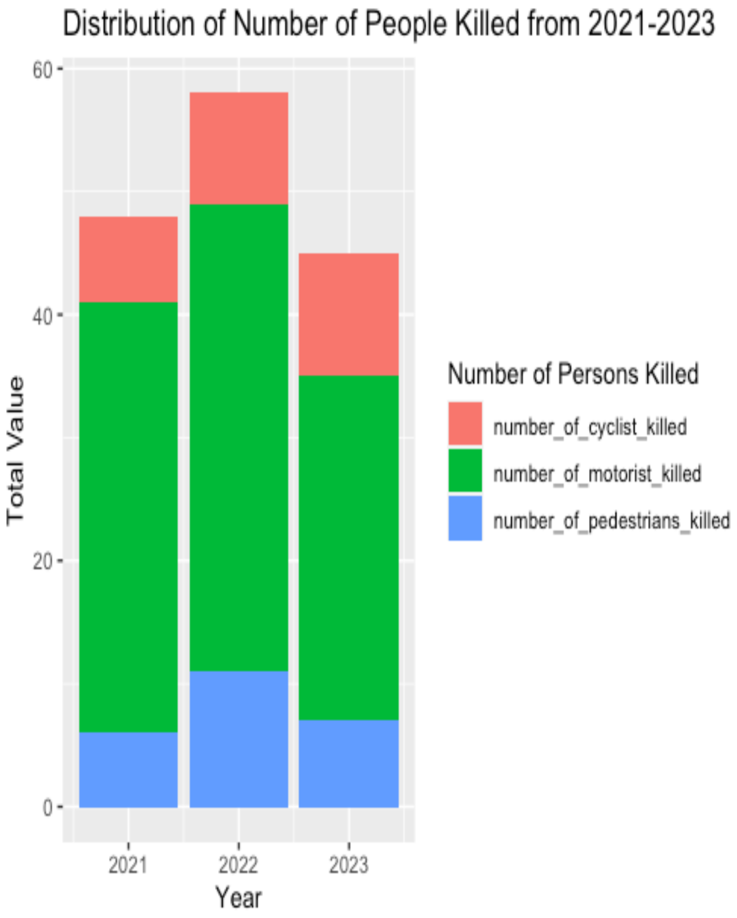
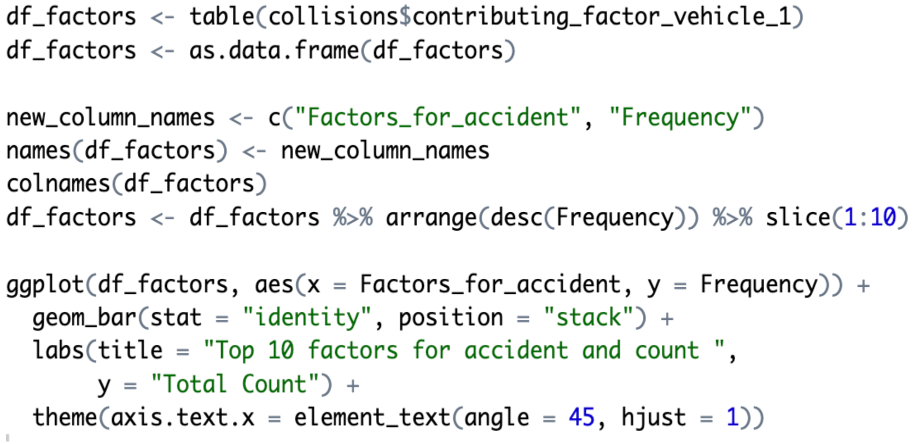


Fig-6: Distribution of People killed due to Accidents



```
df_killed <- collisions %>% group_by(Year)
%>%
  summarise(number_of_persons_killed =
sum(number_of_persons_killed),
            number_of_pedestrians_killed =
sum(number_of_pedestrians_killed),
            number_of_motorist_killed =
sum(number_of_motorist_killed),
            number_of_cyclist_killed =
sum(number_of_cyclist_killed))
# Number of people killed
df_long <- gather(df_killed, key = "Variable",
value = "Value", -Year,-
number_of_persons_killed)
ggplot(df_long, aes(x = Year, y = Value, fill =
Variable)) +
  geom_bar(stat = "identity", position = "stack") +
  labs(title = "Distribution of Number of People
Killed from 2021-2022",
       y = "Total Value", fill = "Number of Persons
Killed")
```

Fig-7: Top 10 Factors that Result in a Crash



```
df_factors <- table(collisions$contributing_factor_vehicle_1)
df_factors <- as.data.frame(df_factors)

new_column_names <- c("Factors_for_accident", "Frequency")
names(df_factors) <- new_column_names
colnames(df_factors)
df_factors <- df_factors %>% arrange(desc(Frequency)) %>% slice(1:10)

ggplot(df_factors, aes(x = Factors_for_accident, y = Frequency)) +
  geom_bar(stat = "identity", position = "stack") +
  labs(title = "Top 10 factors for accident and count ",
       y = "Total Count") +
  theme(axis.text.x = element_text(angle = 45, hjust = 1))
```

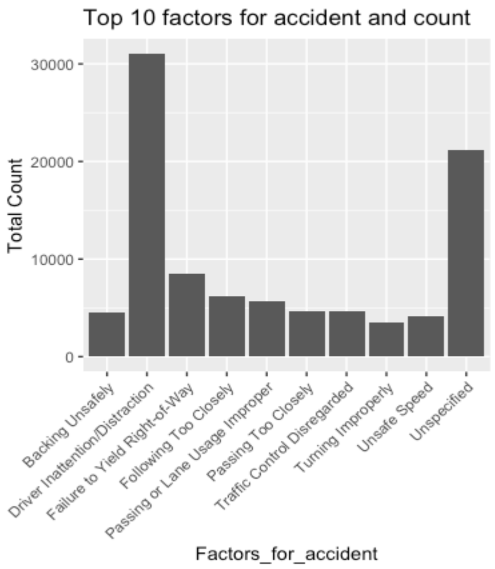
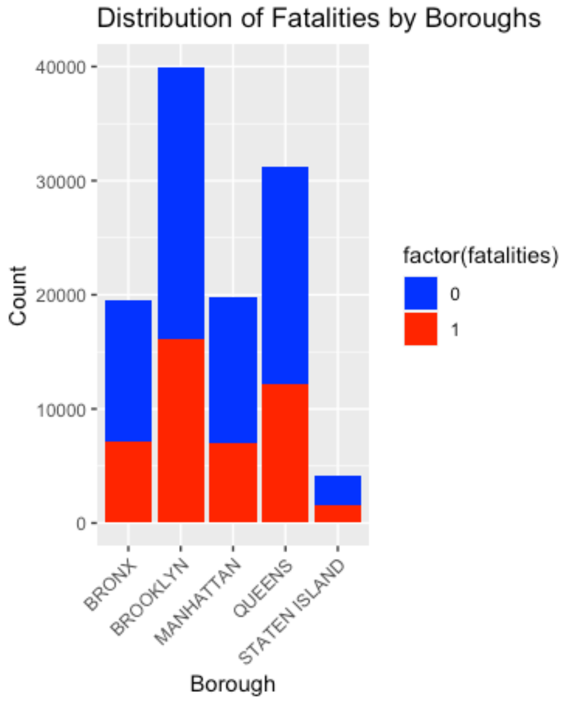
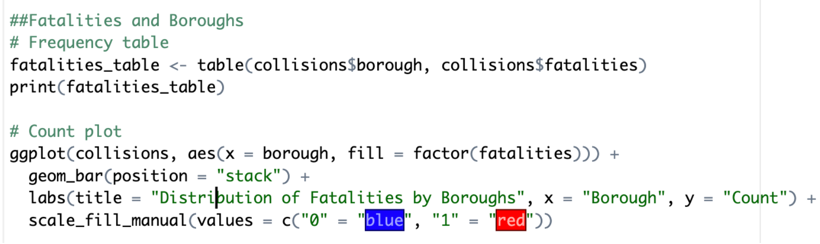


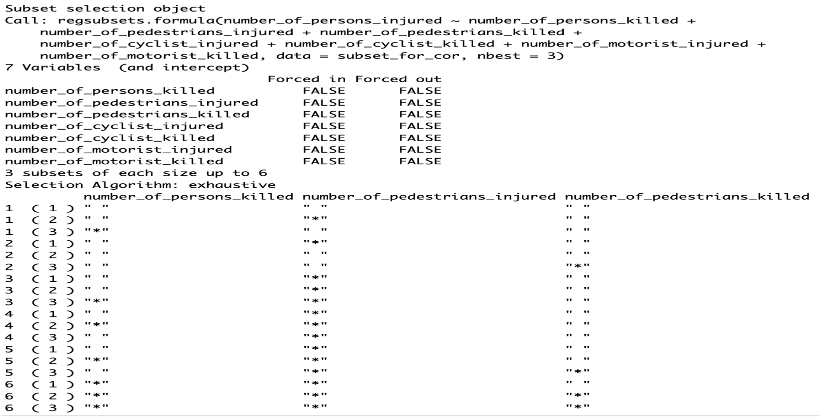
Fig-8: Distribution of Fatalities by Boroughs





```
##Fatalities and Boroughs
# Frequency table
fatalities_table <- table(collisions$borough, collisions$fatalities)
print(fatalities_table)

# Count plot
ggplot(collisions, aes(x = borough, fill = factor(fatalities))) +
  geom_bar(position = "stack") +
  labs(title = "Distribution of Fatalities by Boroughs", x = "Borough", y = "Count") +
  scale_fill_manual(values = c("0" = "blue", "1" = "red"))
```

Fig-9: Subset Model



```
Subset selection object
Call: regsubsets.formula(number_of_persons_injured ~ number_of_persons_killed +
    number_of_pedestrians_injured + number_of_pedestrians_killed +
    number_of_cyclist_injured + number_of_cyclist_killed + number_of_motorist_injured +
    number_of_motorist_killed, data = subset_for_cor, nbest = 3)
7 Variables  (and intercept)
                              Forced in Forced out
number_of_persons_killed          FALSE      FALSE
number_of_pedestrians_injured     FALSE      FALSE
number_of_pedestrians_killed      FALSE      FALSE
number_of_cyclist_injured         FALSE      FALSE
number_of_cyclist_killed          FALSE      FALSE
number_of_motorist_injured        FALSE      FALSE
number_of_motorist_killed         FALSE      FALSE
3 subsets of each size up to 6
Selection Algorithm: exhaustive
         number_of_persons_killed number_of_pedestrians_injured number_of_pedestrians_killed
1  ( 1 ) " "                      " "                            " "
1  ( 2 ) " "                      "*"                            " "
1  ( 3 ) "*"                      " "                            " "
2  ( 1 ) " "                      "*"                            " "
2  ( 2 ) " "                      " "                            " "
2  ( 3 ) " "                      " "                            "*"
3  ( 1 ) " "                      "*"                            " "
3  ( 2 ) " "                      "*"                            " "
3  ( 3 ) "*"                      "*"                            " "
4  ( 1 ) " "                      "*"                            " "
4  ( 2 ) "*"                      "*"                            " "
4  ( 3 ) " "                      "*"                            " "
5  ( 1 ) " "                      "*"                            " "
5  ( 2 ) "*"                      "*"                            " "
5  ( 3 ) " "                      "*"                            "*"
6  ( 1 ) "*"                      "*"                            " "
6  ( 2 ) "*"                      "*"                            "*"
6  ( 3 ) "*"                      "*"                            "*"
```

Fig-10: Multiple Regression Model

```
Call:
lm(formula = number_of_persons_injured ~ number_of_pedestrians_injured +
    number_of_cyclist_injured + number_of_motorist_injured +
    number_of_persons_killed, data = subset_for_cor)

Residuals:
     Min       1Q   Median       3Q      Max 
-0.04481 -0.04481 -0.04481 -0.02474  2.95519 

Coefficients:
                               Estimate Std. Error  t value             Pr(>|t|)    
(Intercept)                   0.0448106  0.0006460   69.363 <0.0000000000000002 ***
number_of_pedestrians_injured 0.9876606  0.0061066  161.735 <0.0000000000000002 ***
number_of_cyclist_injured     0.9601906  0.0021100  455.073 <0.0000000000000002 ***
number_of_motorist_injured    0.9799302  0.0006554 1495.186 <0.0000000000000002 ***
number_of_persons_killed     -0.0218612  0.0134440   -1.626                0.104    
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 0.187 on 114678 degrees of freedom
Multiple R-squared:  0.9532,	Adjusted R-squared:  0.9532 
F-statistic: 5.837e+05 on 4 and 114678 DF,  p-value: < 0.00000000000000022
```

Fig-11: Residuals Vs Fitted

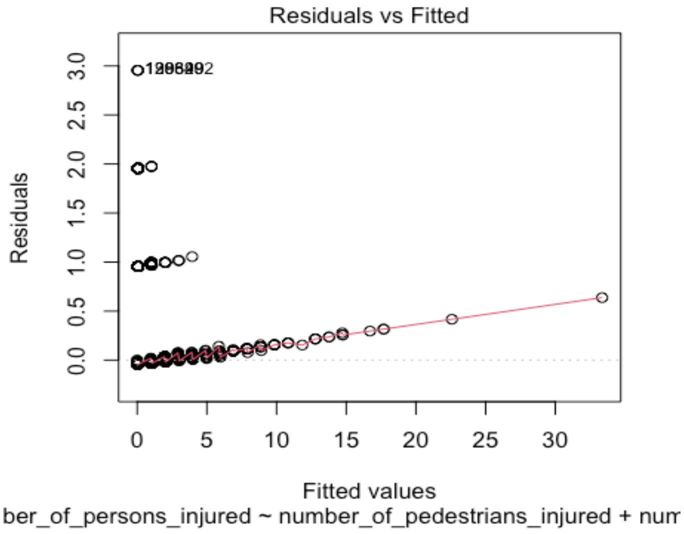


Fig-12: Q-Q Plot

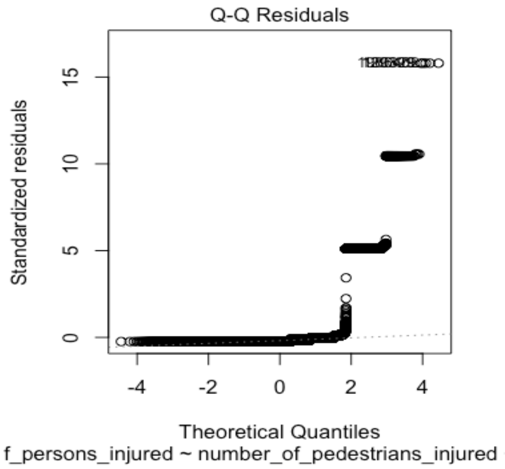


Fig-13: Scale-Location Plot:

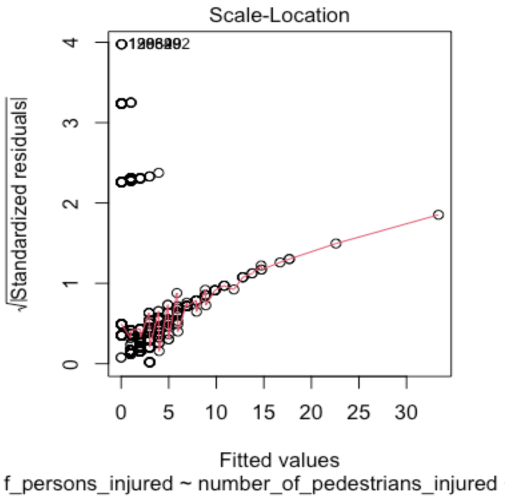


Fig-14: Residuals vs Leverage:

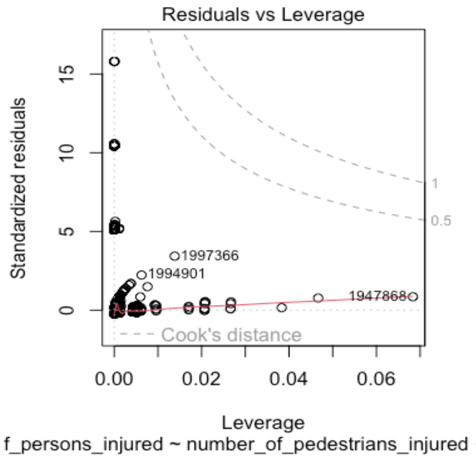
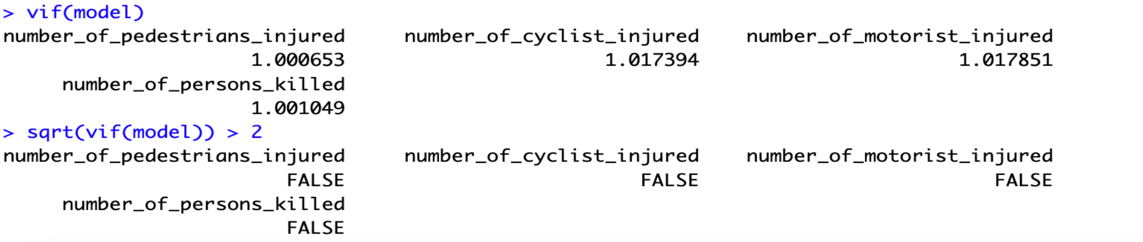
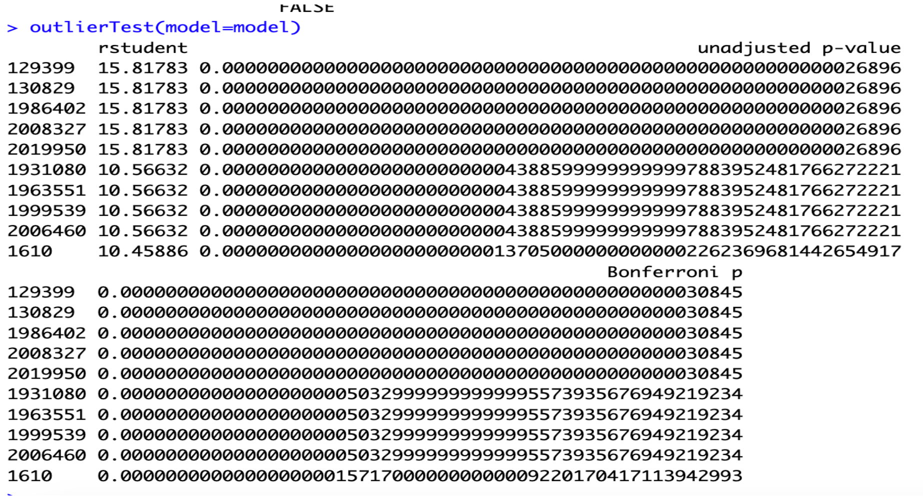


Fig-15: Multi-Collinearity



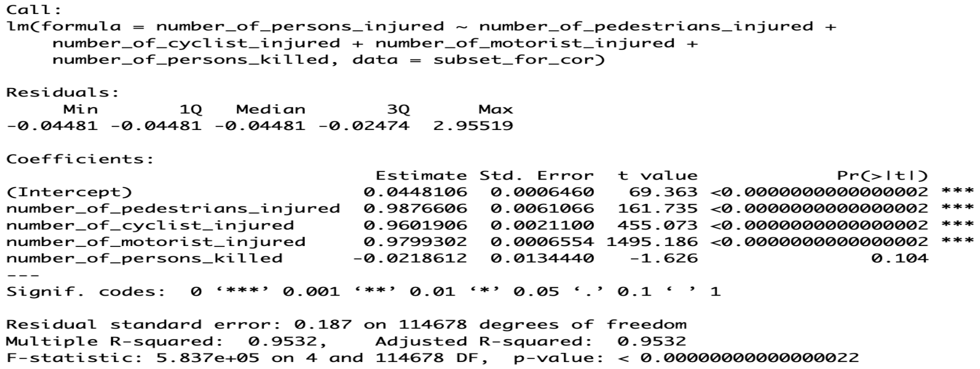
```
> vif(model)
number_of_pedestrians_injured      number_of_cyclist_injured     number_of_motorist_injured
                     1.000653                       1.017394                       1.017851
     number_of_persons_killed
                     1.001049
> sqrt(vif(model)) > 2
number_of_pedestrians_injured      number_of_cyclist_injured     number_of_motorist_injured
                        FALSE                          FALSE                          FALSE
     number_of_persons_killed
                        FALSE
```

Fig-16: Assessing Outliers



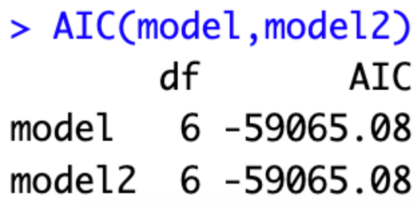
```
                            FALSE
> outlierTest(model=model)
        rstudent                                            unadjusted p-value
129399  15.81783 0.00000000000000000000000000000000000000000000000000000000026896
130829  15.81783 0.00000000000000000000000000000000000000000000000000000000026896
1986402 15.81783 0.00000000000000000000000000000000000000000000000000000000026896
2008327 15.81783 0.00000000000000000000000000000000000000000000000000000000026896
2019950 15.81783 0.00000000000000000000000000000000000000000000000000000000026896
1931080 10.56632 0.000000000000000000000000043885999999999997883952481766272221
1963551 10.56632 0.000000000000000000000000043885999999999997883952481766272221
1999539 10.56632 0.000000000000000000000000043885999999999997883952481766272221
2006460 10.56632 0.000000000000000000000000043885999999999997883952481766272221
1610    10.45886 0.000000000000000000000000137050000000000002262369681442654917
                                                    Bonferroni p
129399  0.0000000000000000000000000000000000000000000000000030845
130829  0.0000000000000000000000000000000000000000000000000030845
1986402 0.0000000000000000000000000000000000000000000000000030845
2008327 0.0000000000000000000000000000000000000000000000000030845
2019950 0.0000000000000000000000000000000000000000000000000030845
1931080 0.000000000000000000005032999999999995573935676949219234
1963551 0.000000000000000000005032999999999995573935676949219234
1999539 0.000000000000000000005032999999999995573935676949219234
2006460 0.000000000000000000005032999999999995573935676949219234
1610    0.0000000000000000000157170000000000009220170417113942993
```

Fig-17: Regression Model-2



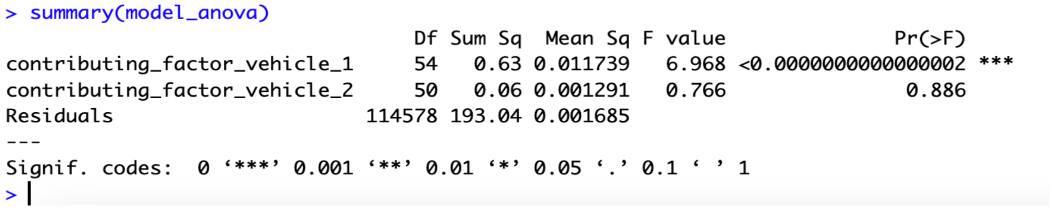
```
Call:
lm(formula = number_of_persons_injured ~ number_of_pedestrians_injured + 
    number_of_cyclist_injured + number_of_motorist_injured + 
    number_of_persons_killed, data = subset_for_cor)

Residuals:
     Min       1Q   Median       3Q      Max 
-0.04481 -0.04481 -0.04481 -0.02474  2.95519 

Coefficients:
                               Estimate Std. Error  t value            Pr(>|t|)    
(Intercept)                   0.0448106  0.0006460   69.363 <0.0000000000000002 ***
number_of_pedestrians_injured 0.9876606  0.0061066  161.735 <0.0000000000000002 ***
number_of_cyclist_injured     0.9601906  0.0021100  455.073 <0.0000000000000002 ***
number_of_motorist_injured    0.9799302  0.0006554 1495.186 <0.0000000000000002 ***
number_of_persons_killed     -0.0218612  0.0134440   -1.626               0.104    
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 0.187 on 114678 degrees of freedom
Multiple R-squared:  0.9532,    Adjusted R-squared:  0.9532 
F-statistic: 5.837e+05 on 4 and 114678 DF,  p-value: < 0.00000000000000022
```

Fig-18: Model Comparison



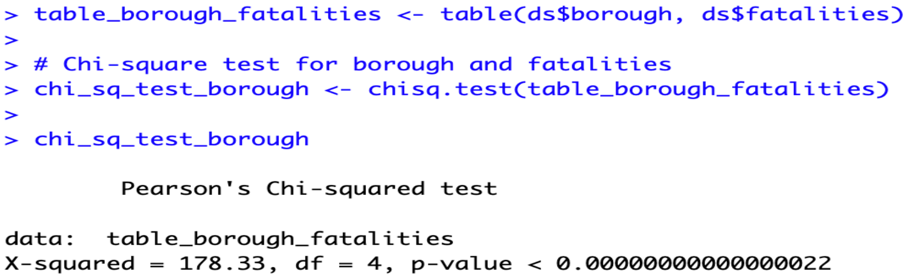
```
> AIC(model,model2)
       df       AIC
model   6 -59065.08
model2  6 -59065.08
```

Fig-19: ANOVA



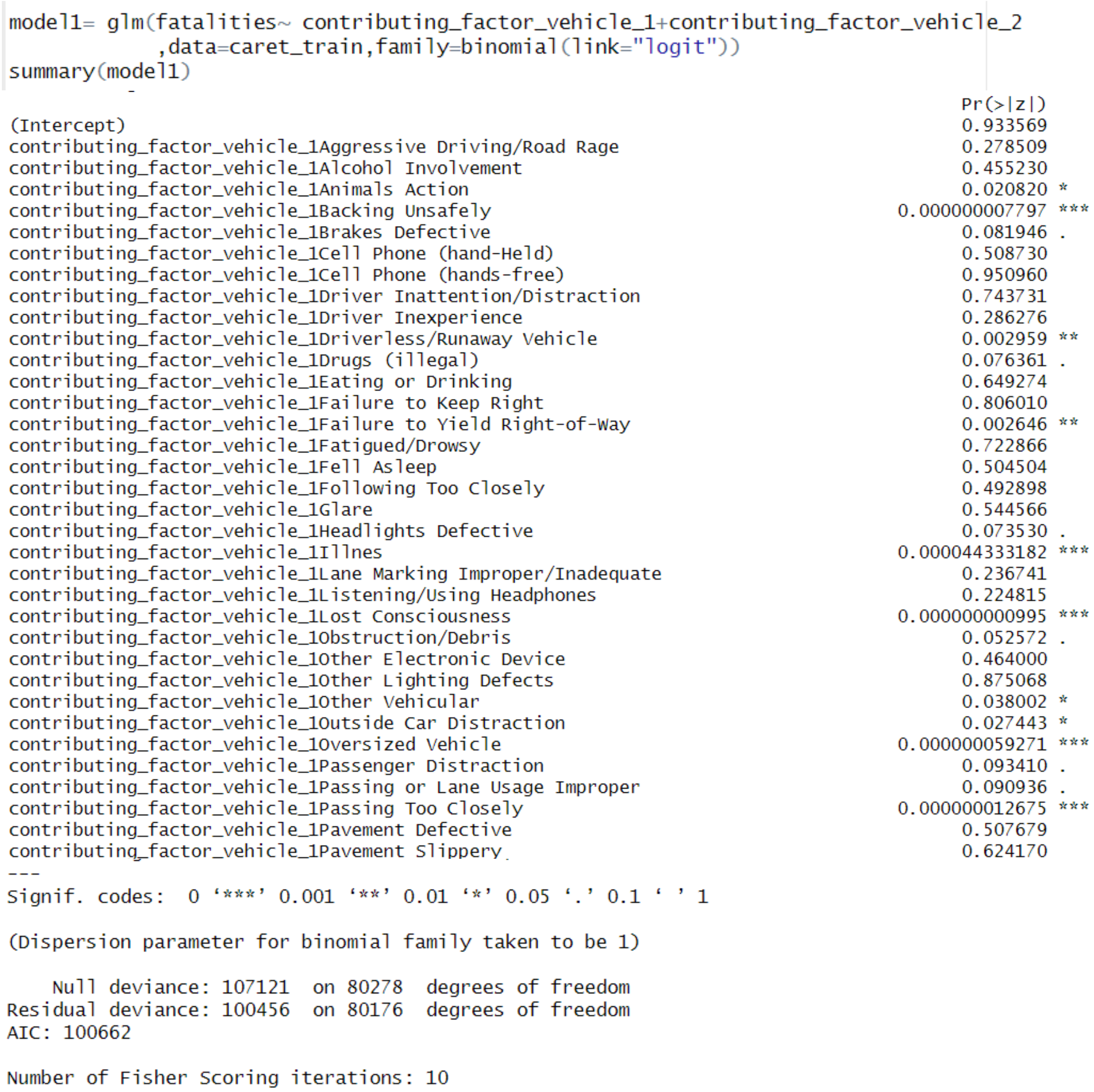
```
> summary(model_anova)
                                Df Sum Sq  Mean Sq F value                Pr(>F)    
contributing_factor_vehicle_1     54   0.63 0.011739   6.968 <0.0000000000000002 ***
contributing_factor_vehicle_2     50   0.06 0.001291   0.766                 0.886    
Residuals                     114578 193.04 0.001685                                 
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1
> 
```

Fig-20: Chi-Square



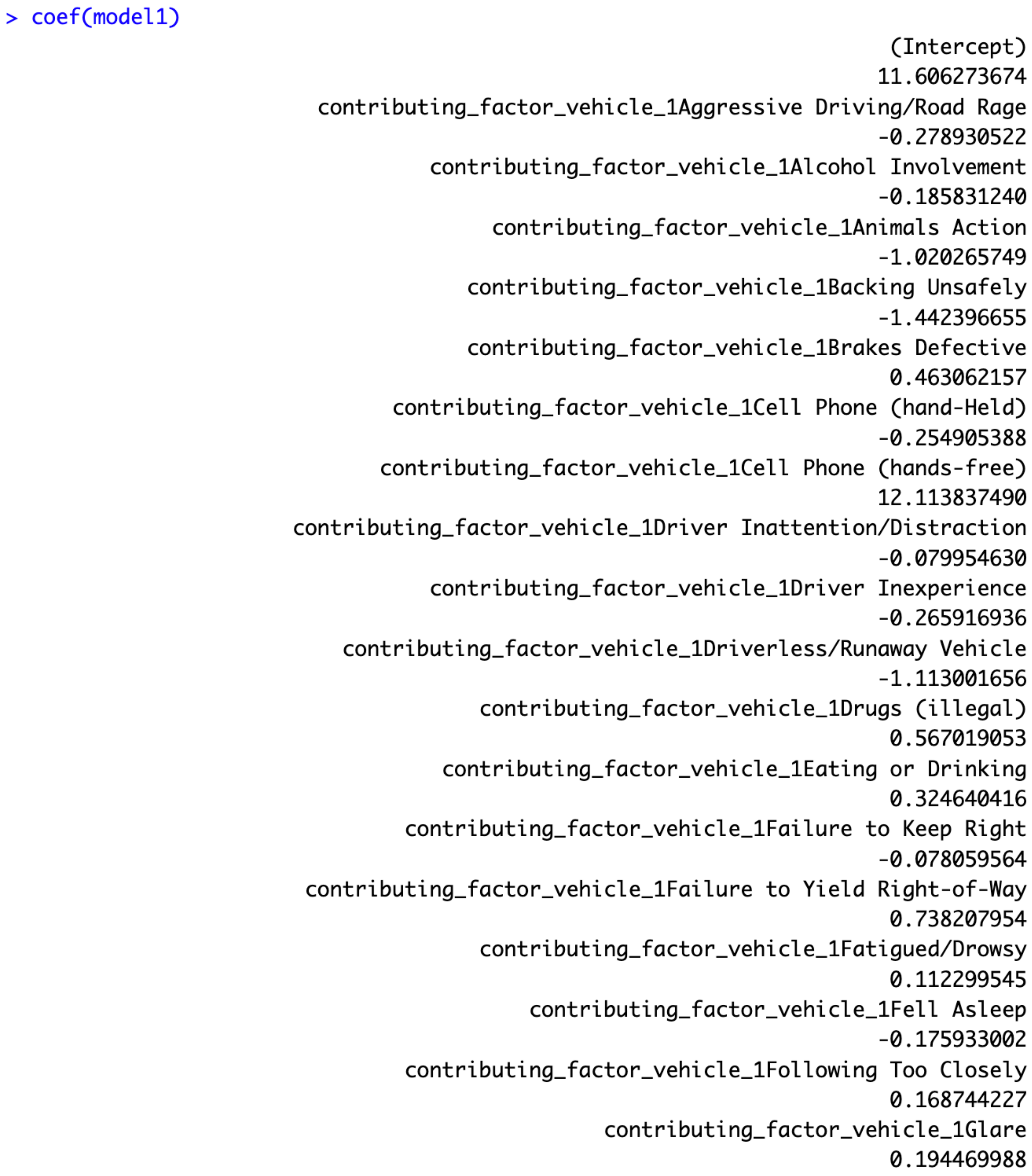
```
> table_borough_fatalities <- table(ds$borough, ds$fatalities)
> 
> # Chi-square test for borough and fatalities
> chi_sq_test_borough <- chisq.test(table_borough_fatalities)
> 
> chi_sq_test_borough

	Pearson's Chi-squared test

data:  table_borough_fatalities
X-squared = 178.33, df = 4, p-value < 0.00000000000000022
```

Fig-21: Summary of Model1 and coefficients



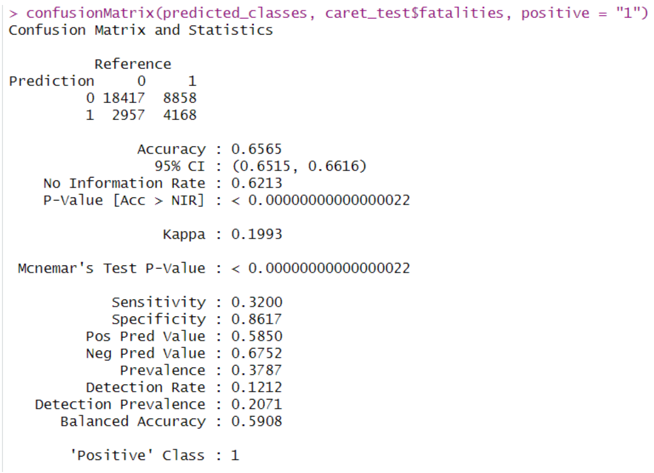
```
model1= glm(fatalities~ contributing_factor_vehicle_1+contributing_factor_vehicle_2
            ,data=caret_train,family=binomial(link="logit"))
summary(model1)
```

```
                                                                        Pr(>|z|)    
(Intercept)                                                             0.933569    
contributing_factor_vehicle_1Aggressive Driving/Road Rage               0.278509    
contributing_factor_vehicle_1Alcohol Involvement                        0.455230    
contributing_factor_vehicle_1Animals Action                             0.020820 *  
contributing_factor_vehicle_1Backing Unsafely                      0.000000007797 ***
contributing_factor_vehicle_1Brakes Defective                           0.081946 .  
contributing_factor_vehicle_1Cell Phone (hand-Held)                     0.508730    
contributing_factor_vehicle_1Cell Phone (hands-free)                    0.950960    
contributing_factor_vehicle_1Driver Inattention/Distraction             0.743731    
contributing_factor_vehicle_1Driver Inexperience                        0.286276    
contributing_factor_vehicle_1Driverless/Runaway Vehicle                 0.002959 ** 
contributing_factor_vehicle_1Drugs (illegal)                            0.076361 .  
contributing_factor_vehicle_1Eating or Drinking                         0.649274    
contributing_factor_vehicle_1Failure to Keep Right                      0.806010    
contributing_factor_vehicle_1Failure to Yield Right-of-Way              0.002646 ** 
contributing_factor_vehicle_1Fatigued/Drowsy                            0.722866    
contributing_factor_vehicle_1Fell Asleep                                0.504504    
contributing_factor_vehicle_1Following Too Closely                      0.492898    
contributing_factor_vehicle_1Glare                                      0.544566    
contributing_factor_vehicle_1Headlights Defective                       0.073530 .  
contributing_factor_vehicle_1Illnes                                0.000044333182 ***
contributing_factor_vehicle_1Lane Marking Improper/Inadequate           0.236741    
contributing_factor_vehicle_1Listening/Using Headphones                 0.224815    
contributing_factor_vehicle_1Lost Consciousness                    0.000000000995 ***
contributing_factor_vehicle_1Obstruction/Debris                         0.052572 .  
contributing_factor_vehicle_1Other Electronic Device                    0.464000    
contributing_factor_vehicle_1Other Lighting Defects                     0.875068    
contributing_factor_vehicle_1Other Vehicular                            0.038002 *  
contributing_factor_vehicle_1Outside Car Distraction                    0.027443 *  
contributing_factor_vehicle_1Oversized Vehicle                     0.000000059271 ***
contributing_factor_vehicle_1Passenger Distraction                      0.093410 .  
contributing_factor_vehicle_1Passing or Lane Usage Improper             0.090936 .  
contributing_factor_vehicle_1Passing Too Closely                   0.000000012675 ***
contributing_factor_vehicle_1Pavement Defective                         0.507679    
contributing_factor_vehicle_1Pavement Slippery                          0.624170    
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 107121  on 80278  degrees of freedom
Residual deviance: 100456  on 80176  degrees of freedom
AIC: 100662

Number of Fisher Scoring iterations: 10
```



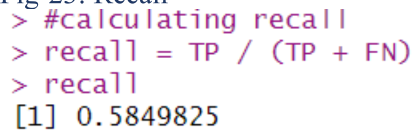
```
> coef(model1)
                                                (Intercept) 
                                               11.606273674 
  contributing_factor_vehicle_1Aggressive Driving/Road Rage 
                                               -0.278930522 
           contributing_factor_vehicle_1Alcohol Involvement 
                                               -0.185831240 
                contributing_factor_vehicle_1Animals Action 
                                               -1.020265749 
              contributing_factor_vehicle_1Backing Unsafely 
                                               -1.442396655 
              contributing_factor_vehicle_1Brakes Defective 
                                                0.463062157 
        contributing_factor_vehicle_1Cell Phone (hand-Held) 
                                               -0.254905388 
       contributing_factor_vehicle_1Cell Phone (hands-free) 
                                               12.113837490 
contributing_factor_vehicle_1Driver Inattention/Distraction 
                                               -0.079954630 
           contributing_factor_vehicle_1Driver Inexperience 
                                               -0.265916936 
    contributing_factor_vehicle_1Driverless/Runaway Vehicle 
                                               -1.113001656 
               contributing_factor_vehicle_1Drugs (illegal) 
                                                0.567019053 
            contributing_factor_vehicle_1Eating or Drinking 
                                                0.324640416 
         contributing_factor_vehicle_1Failure to Keep Right 
                                               -0.078059564 
 contributing_factor_vehicle_1Failure to Yield Right-of-Way 
                                                0.738207954 
               contributing_factor_vehicle_1Fatigued/Drowsy 
                                                0.112299545 
                   contributing_factor_vehicle_1Fell Asleep 
                                               -0.175933002 
         contributing_factor_vehicle_1Following Too Closely 
                                                0.168744227 
                         contributing_factor_vehicle_1Glare 
                                                0.194469988 
```

Fig-22: Confusion Matrix



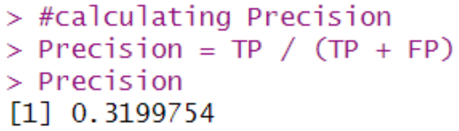
```
> confusionMatrix(predicted_classes, caret_test$fatalities, positive = "1")
Confusion Matrix and Statistics

          Reference
Prediction     0     1
         0 18417  8858
         1  2957  4168
                                               
               Accuracy : 0.6565               
                 95% CI : (0.6515, 0.6616)     
    No Information Rate : 0.6213               
    P-Value [Acc > NIR] : < 0.00000000000000022
                                               
                  Kappa : 0.1993               
                                               
 Mcnemar's Test P-Value : < 0.00000000000000022
                                               
            Sensitivity : 0.3200               
            Specificity : 0.8617               
         Pos Pred Value : 0.5850               
         Neg Pred Value : 0.6752               
             Prevalence : 0.3787               
         Detection Rate : 0.1212               
   Detection Prevalence : 0.2071               
      Balanced Accuracy : 0.5908               
                                               
       'Positive' Class : 1                    
```

Fig-23: Recall



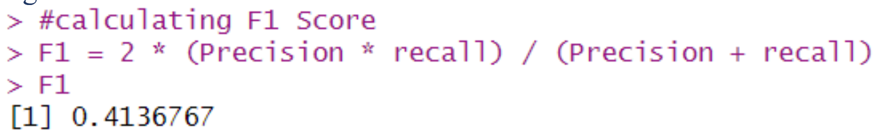
```
> #calculating recall
> recall = TP / (TP + FN)
> recall
[1] 0.5849825
```

Fig-24: Precision



```
> #calculating Precision
> Precision = TP / (TP + FP)
> Precision
[1] 0.3199754
```

Fig-25: F1 score



```
> #calculating F1 Score
> F1 = 2 * (Precision * recall) / (Precision + recall)
> F1
[1] 0.4136767
```

Fig-27: ROC

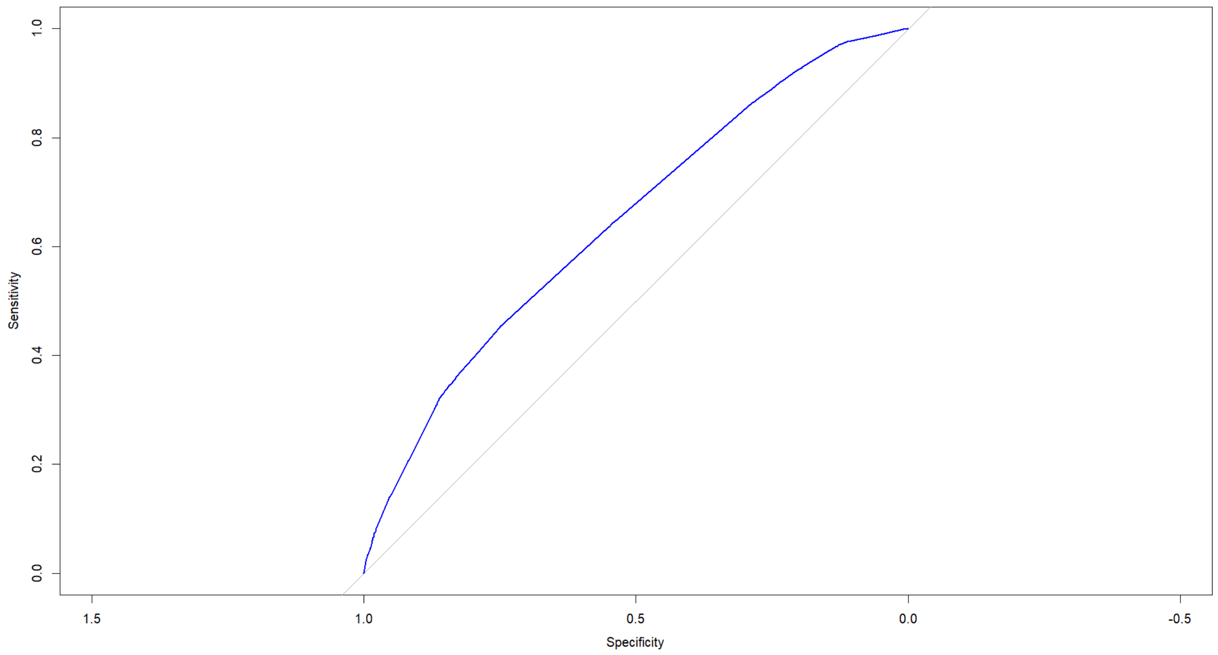
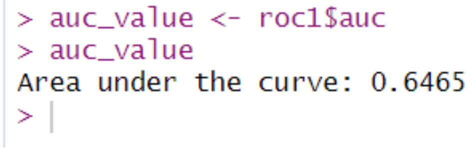
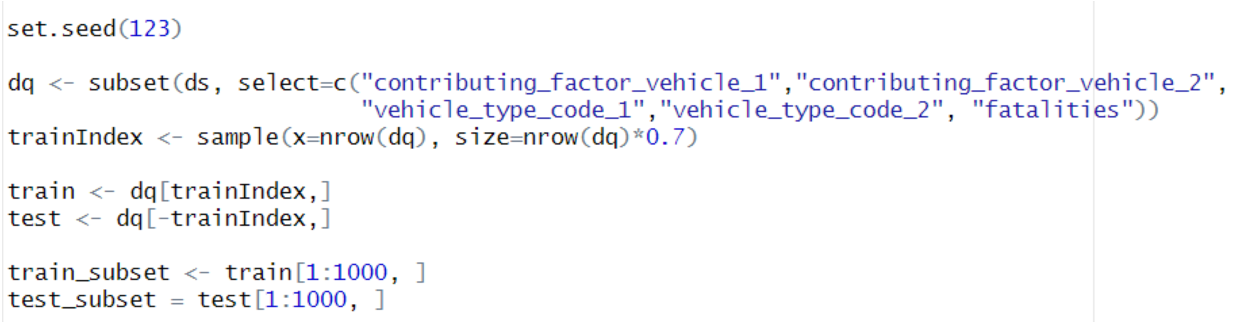


Fig-26: AUC



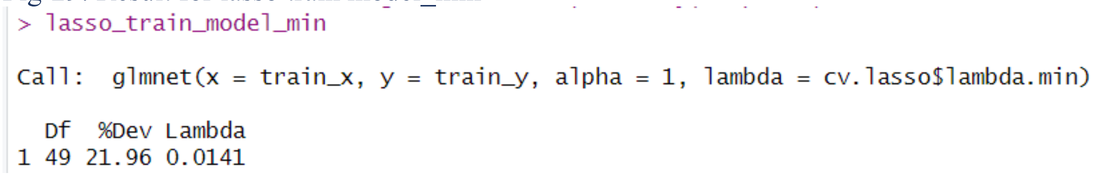
```
> auc_value <- roc1$auc
> auc_value
Area under the curve: 0.6465
> 
```

Fig-28: Splitting the dataset



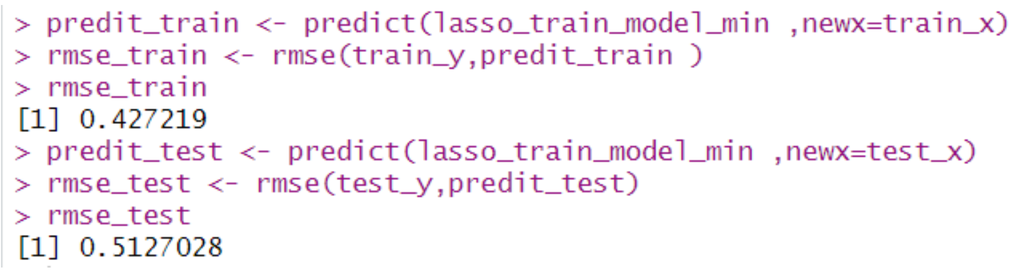
```
set.seed(123)

dq <- subset(ds, select=c("contributing_factor_vehicle_1","contributing_factor_vehicle_2",
                          "vehicle_type_code_1","vehicle_type_code_2", "fatalities"))
trainIndex <- sample(x=nrow(dq), size=nrow(dq)*0.7)

train <- dq[trainIndex,]
test <- dq[-trainIndex,]

train_subset <- train[1:1000, ]
test_subset = test[1:1000, ]
```

Fig-29: Result for lasso train model_min



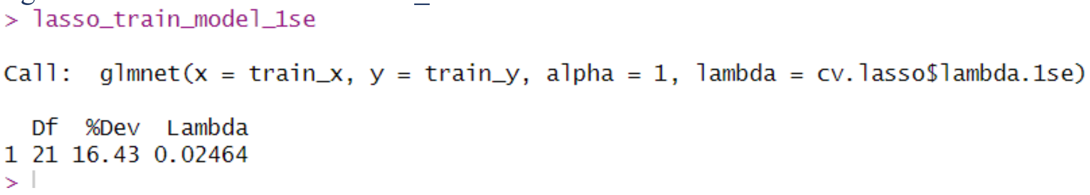
```
> lasso_train_model_min

Call:  glmnet(x = train_x, y = train_y, alpha = 1, lambda = cv.lasso$lambda.min)

  Df  %Dev Lambda
1 49 21.96 0.0141
```

Fig-30: RMSE values



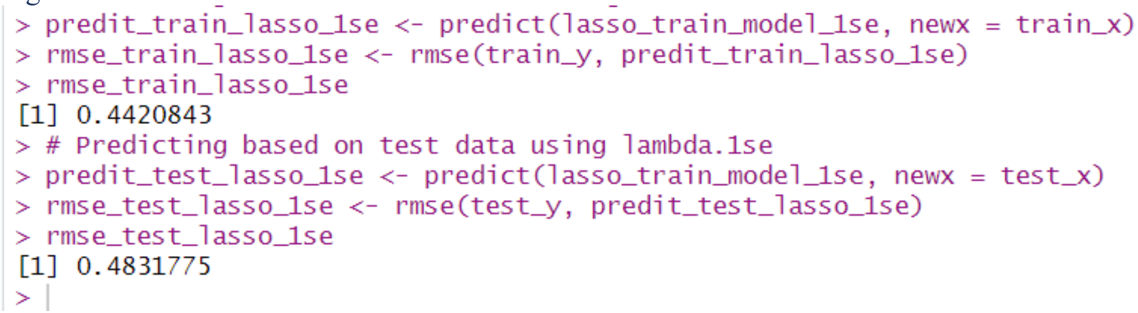
```
> predit_train <- predict(lasso_train_model_min ,newx=train_x)
> rmse_train <- rmse(train_y,predit_train )
> rmse_train
[1] 0.427219
> predit_test <- predict(lasso_train_model_min ,newx=test_x)
> rmse_test <- rmse(test_y,predit_test)
> rmse_test
[1] 0.5127028
```

Fig-31: Result for lasso train model_lse



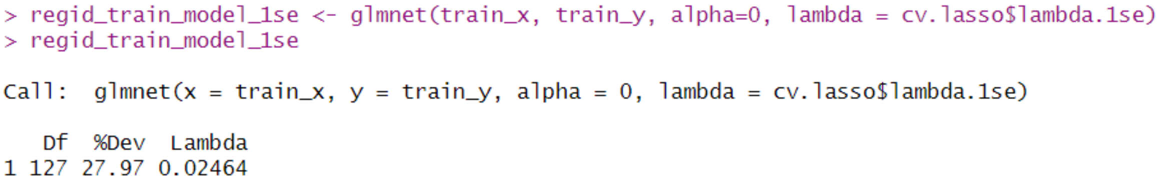
```
> lasso_train_model_1se

Call:  glmnet(x = train_x, y = train_y, alpha = 1, lambda = cv.lasso$lambda.1se)

  Df  %Dev  Lambda
1 21 16.43 0.02464
> 
```

Fig-32: RMSE values



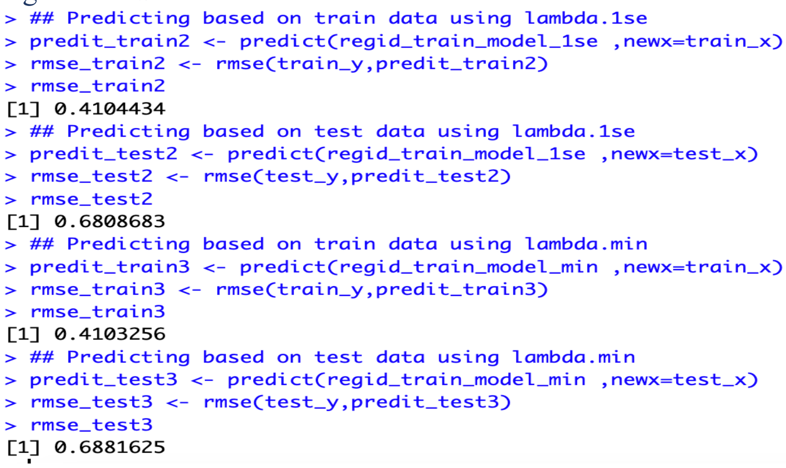
```
> predit_train_lasso_1se <- predict(lasso_train_model_1se, newx = train_x)
> rmse_train_lasso_1se <- rmse(train_y, predit_train_lasso_1se)
> rmse_train_lasso_1se
[1] 0.4420843
> # Predicting based on test data using lambda.1se
> predit_test_lasso_1se <- predict(lasso_train_model_1se, newx = test_x)
> rmse_test_lasso_1se <- rmse(test_y, predit_test_lasso_1se)
> rmse_test_lasso_1se
[1] 0.4831775
> 
```

Fig-33: Result for lasso train model_lse



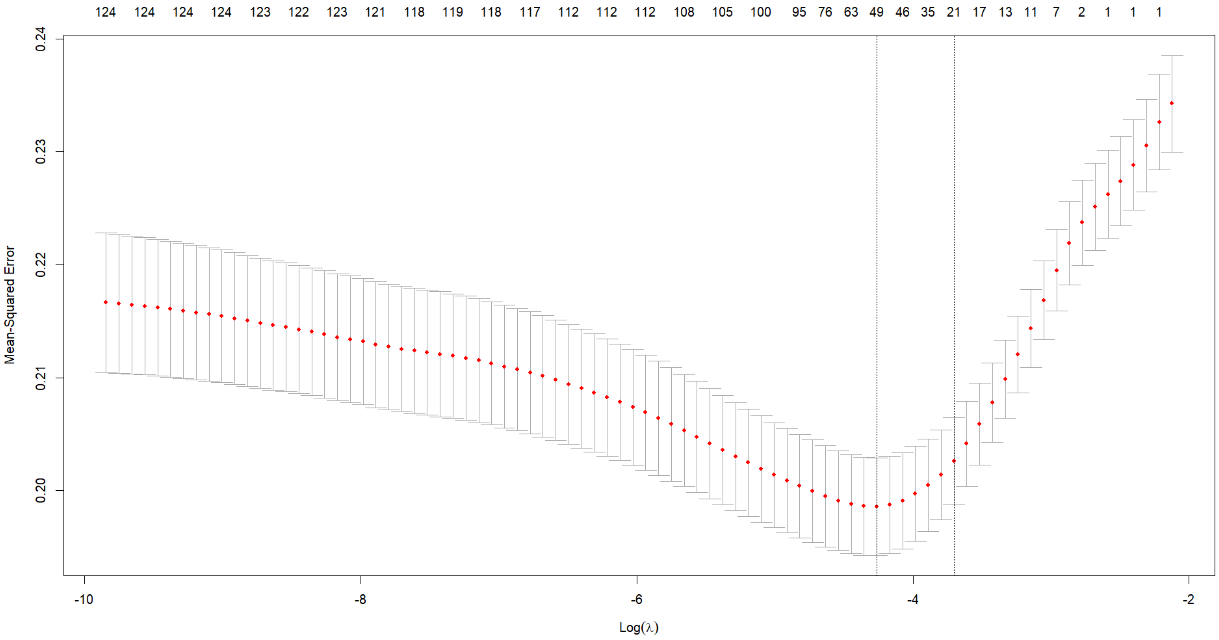
```
> regid_train_model_1se <- glmnet(train_x, train_y, alpha=0, lambda = cv.lasso$lambda.1se)
> regid_train_model_1se

Call:  glmnet(x = train_x, y = train_y, alpha = 0, lambda = cv.lasso$lambda.1se)

   Df  %Dev  Lambda
1 127 27.97 0.02464
```

Fig-34: RMSE values



```
> ## Predicting based on train data using lambda.1se
> predit_train2 <- predict(regid_train_model_1se ,newx=train_x)
> rmse_train2 <- rmse(train_y,predit_train2)
> rmse_train2
[1] 0.4104434
> ## Predicting based on test data using lambda.1se
> predit_test2 <- predict(regid_train_model_1se ,newx=test_x)
> rmse_test2 <- rmse(test_y,predit_test2)
> rmse_test2
[1] 0.6808683
> ## Predicting based on train data using lambda.min
> predit_train3 <- predict(regid_train_model_min ,newx=train_x)
> rmse_train3 <- rmse(train_y,predit_train3)
> rmse_train3
[1] 0.4103256
> ## Predicting based on test data using lambda.min
> predit_test3 <- predict(regid_train_model_min ,newx=test_x)
> rmse_test3 <- rmse(test_y,predit_test3)
> rmse_test3
[1] 0.6881625
```

Fig-35: plot



124 124 124 124 123 122 123 121 118 119 118 117 112 112 112 108 105 100 95 76 63 49 46 35 21 17 13 11 7 2 1 1 1
Mean-Squared Error
0.20
0.21
0.22
0.23
0.24
-10
-8
-6
-4
-2
Log(λ)

References:

Motor Vehicle Collisions - Crashes. (n.d.):[online]; https://www.kaggle.com/datasets/tush32/motor-vehicle-collisions-crashes/data
NEU: Regularization; [online]: https://northeastern.instructure.com/courses/174682/assignments/2098624

Since.smith: Ridge Regression and the Lasso; [online]: Lab 10 - Ridge Regression and the Lasso in R (smith.edu)

Trevor Hastie, Junyang Qian, Kenneth Tay; (27 March 2023): An Introduction to glmnet; [online]: An Introduction to `glmnet` • glmnet (stanford.edu)

Dr Simonj (10 April 2017): How and when: ridge regression with glmnet; [online]: How and when: ridge regression with glmnet (svbtle.com)

Datacamp (November 2019): Regularization in R Tutorial: Ridge, Lasso and Elastic Net; [online]: Regularization in R Tutorial: Ridge, Lasso & Elastic Net Regression | DataCamp

University of Sidney: Lasso Regression; [online]: 4 Lasso Regression | Machine Learning for Biostatistics (bookdown.org)

Jose M Sallan (17 June 2022): Regularized regression with glmnet; [online]: Regularized regression with glmnet - Jose M Sallan blog (jmsallan.netlify.app)

Kassambara (3 November 2018): Penalized Logistic Regression Essentials in R: Ridge, Lasso and Elastic Net; [online]: Penalized Logistic Regression Essentials in R: Ridge, Lasso and Elastic Net - Articles - STHDA