Final Project Report: Traffic Fatalities of Austin, Texas

Introduction

This project explores possible factors that correlate to higher traffic fatalities in Austin in the years 2013-2019. Studying the factors that contribute to traffic fatalities in Austin, Texas, is of paramount importance, considering the profound implications on public safety, health, and the overall well-being of the community. As one of the fastest-growing metropolitan areas in the United States, Austin has experienced a surge in population, leading to increased vehicular congestion and subsequently higher risks on the road. Understanding the underlying causes of traffic accidents and fatalities is crucial for policymakers, urban planners, and law enforcement authorities to implement effective preventive measures and improve road safety infrastructure.

Data

Traffic fatality data from the years 2013 through 2019 was sourced from data.austintexas.gov.
Features of these data sets include:

- Type of fatality (motor vehicle, pedestrian, or motorcycle)
- Number of fatalities
- Case number
- Location (street address)
- Date
- Day of the week
- Time
- Whether the driver or passenger was killed
- Speeding (yes/no)
- If the driver ran a red light or stop sign
- Driver's license status
- Suspected impairment
- Type of road

Moon phase data was sourced from nasa.gov.
Features of this data includes:

- Month
- Day (number)
- Time
- Moon phase
- Moon age
- Moon diameter
- Moon distance from Earth

- Moon phase category (waxing, waning, gibbous, crescent, etc.)

Steps taken to clean the data:

Clean Moon Phase data

- Load each moon_phase dataset of each year independently as text files. Text files then are read into a table separating columns by tab delimiters and rows by enter delimiters. Label each column. Each annual dataset is mutated alongside the corresponding year and merged together with dplyr row bound functions. Group in order by date and time.

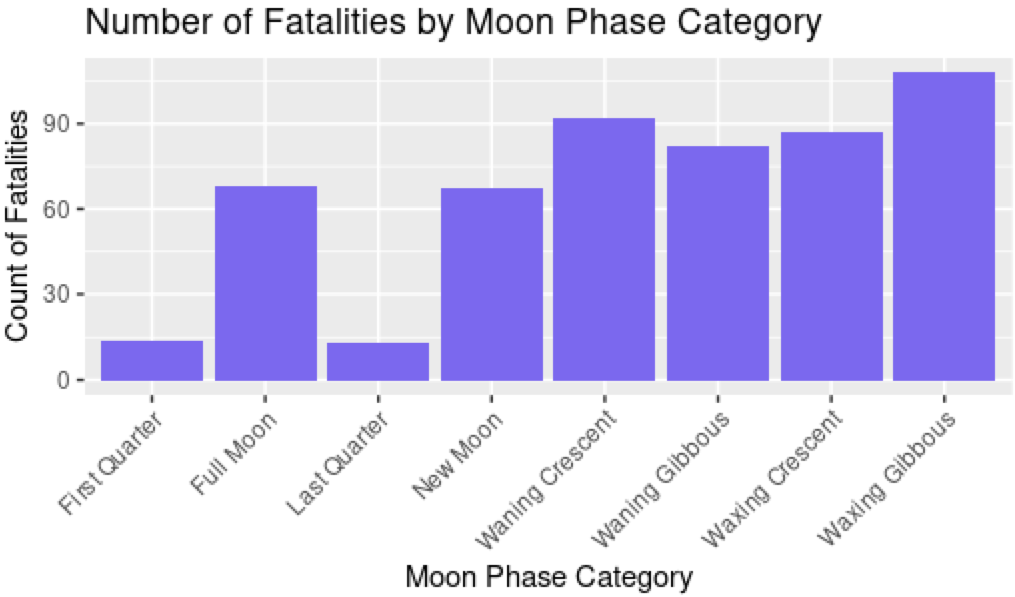
Clean Traffic Fatality data

- Load in traffic_fatality dataset as csv. Remove extraneous columns: “related”, “restraint type”, “x and y coord”, “fail to stop and render aid”, and “Homeless”. Group in order by date and time

Merge Datasets

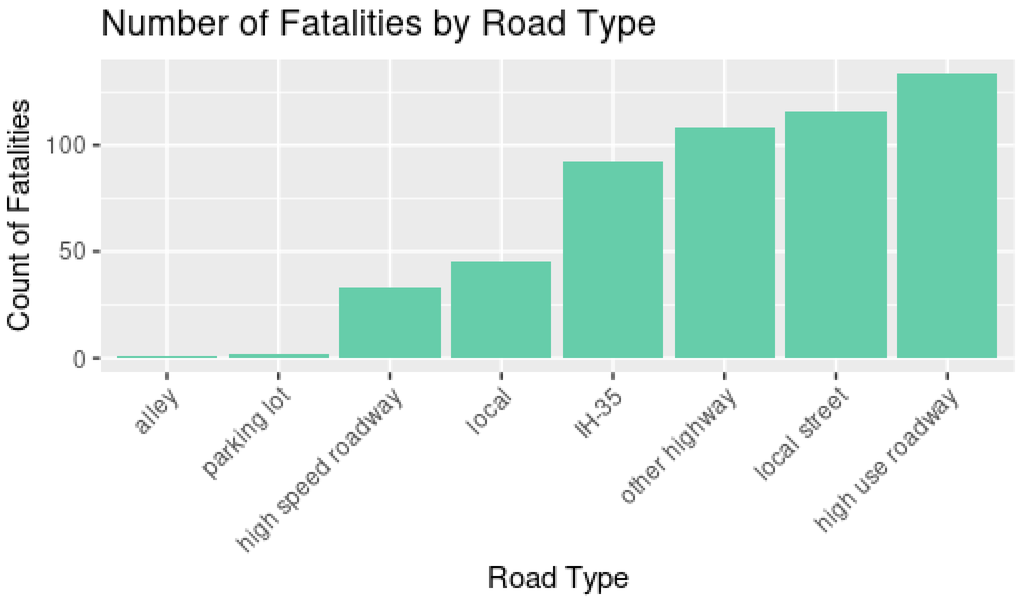
- Merge by common date and time. Ignore and remove unmerged/missing value rows. Remove extraneous “date” and “time” columns. Convert date columns to date format.

Exploratory Analysis



Hypothesis 1: During a waxing gibbous, there are more occurrences of traffic fatalities in Austin, Texas.

```
ggplot(project, aes(x = MoonPhaseCat)) +
  geom_bar(fill = "mediumslateblue") +
  labs(title = "Number of Fatalities by Moon Phase Category", x = "Moon Phase
Category", y = "Count of Fatalities") +
  theme(axis.text.x = element_text(angle = 45, hjust = 1))
```

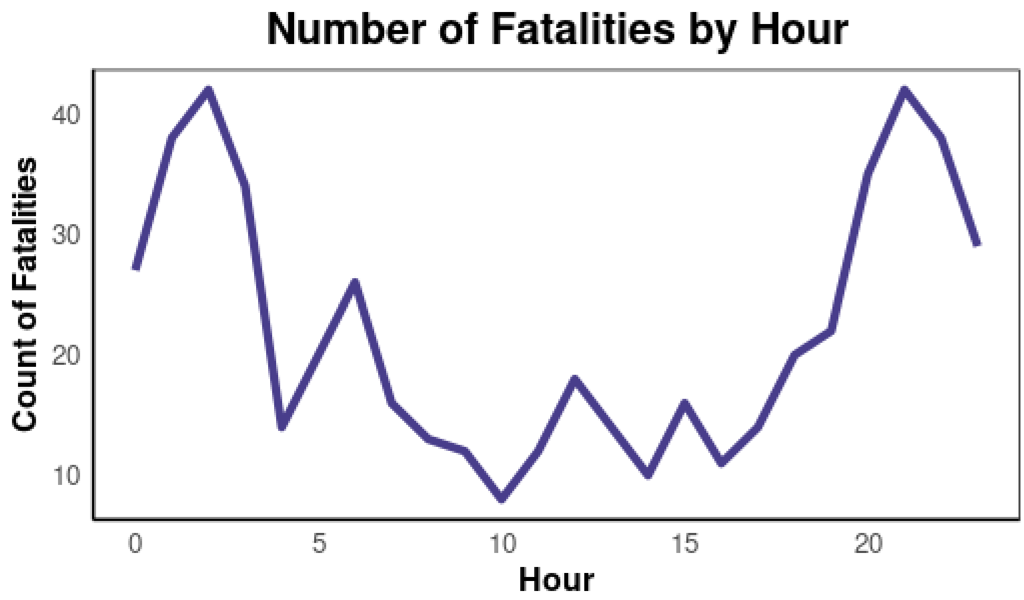


Hypothesis 2: High use roadways have significantly more traffic fatalities than any other roadways.
R-squared value: 0.0231 (low correlation, <0.2)
Adjusted R-squared: -0.005264
Standard Error: 0.2566465

```
project %>%
  group_by(Type.of.road) %>%
  summarise(FatalitiesCount = n()) %>%
  ggplot(aes(x = reorder(Type.of.road, FatalitiesCount), y = FatalitiesCount)) +
  geom_bar(fill = "aquamarine3", stat = "identity") +
  labs(title = "Number of Fatalities by Road Type", x = "Road Type", y = "Count of
Fatalities") +
  theme(axis.text.x = element_text(angle = 45, hjust = 1))
```

#R Statistics:

```
model <- lm(`Number of Fatalities` ~ `Type of road`, data = project)
rsquared <- summary(model)$r.squared
adjusted_rsquared <- summary(model)$adj.r.squared
standard_error <- summary(model)$sigma
cat("R-squared value:", rsquared, "\n")
cat("Adjusted R-squared:", adjusted_rsquared, "\n")
cat("Standard Error:", standard_error, "\n")
```

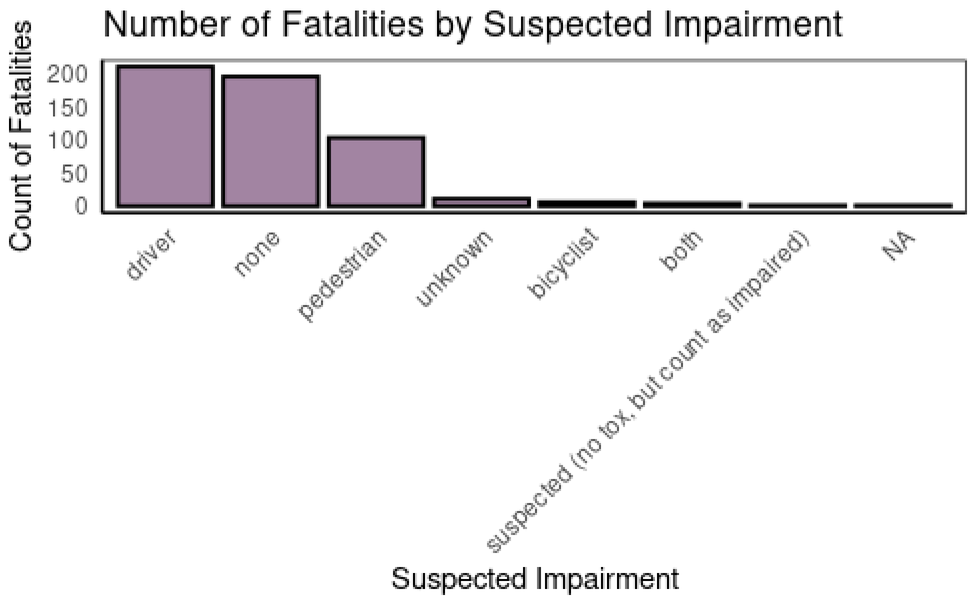


Hypothesis 3: During daylight hours, there are significantly less traffic fatalities than during nighttime hours.
R-squared value: 0.04563 (low correlation)
Adjusted R-squared: 0.0023429
Standard Error: 0.2556736

```
project$Hour <- as.numeric(project$Hour)
ggplot(project, aes(x = Hour)) +
  geom_line(stat = "count", color = "darkslateblue", size = 1.5) +
  labs(title = "Number of Fatalities by Hour", x = "Hour", y = "Count of
Fatalities") +
  theme_minimal() +
  theme(
    panel.grid.major = element_blank(),
    panel.grid.minor = element_blank(),
    panel.background = element_rect(fill = "white"),
    axis.line = element_line(color = "black"),
    axis.text = element_text(size = 10),
    axis.title = element_text(size = 12, face = "bold"),
    plot.title = element_text(size = 16, face = "bold", hjust = 0.5),
    legend.position = "none"  # Remove legend if not needed
  )
```

#R statistics:

```
model <- lm(`Number of Fatalities` ~ `Hour`, data = project)
rsquared <- summary(model)$r.squared
adjusted_rsquared <- summary(model)$adj.r.squared
standard_error <- summary(model)$sigma
cat("R-squared value:", rsquared, "\n")
cat("Adjusted R-squared:", adjusted_rsquared, "\n")
cat("Standard Error:", standard_error, "\n")
```

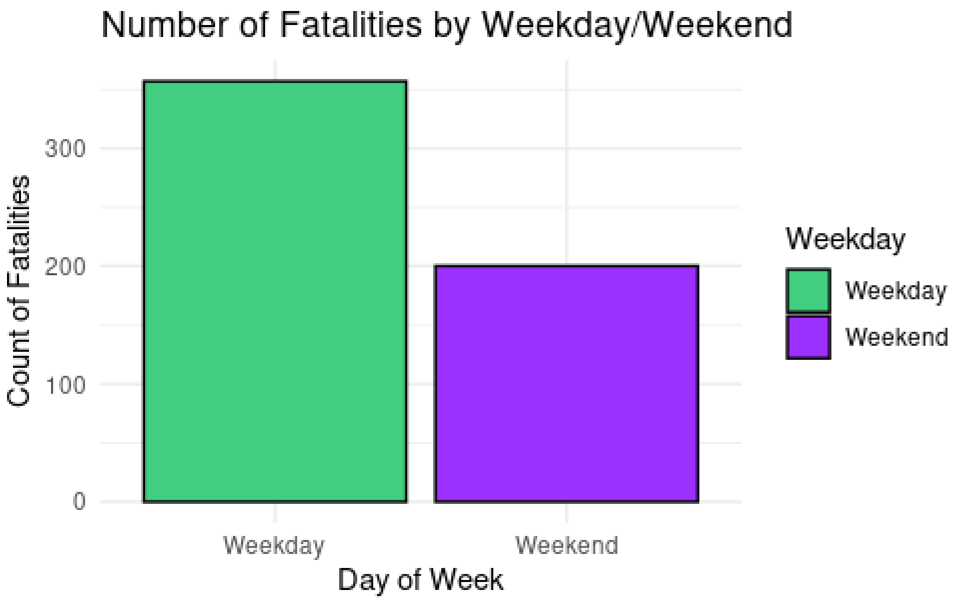


Hypothesis 4: When the driver is suspected to be impaired, more traffic fatalities occur.
R squared value: 0.077680 (low correlation)
Adjusted R-squared: 0.0357566
Standard Error: 0.251584

```
project$Suspected.Impairment <- fct_reorder(project$Suspected.Impairment,
project$Number.of.Fatalities, .fun = sum, .desc = TRUE)

ggplot(project, aes(x = Suspected.Impairment)) +
  geom_bar(fill = "plum4", color = "black", size = 0.7, alpha = 0.8) +
  labs(title = "Number of Fatalities by Suspected Impairment", x = "Suspected
Impairment", y = "Count of Fatalities") +
  theme_minimal() +
  theme(
    axis.text.x = element_text(angle = 45, hjust = 1, vjust = 1),
    panel.grid.major = element_blank(),
    panel.grid.minor = element_blank(),
    panel.background = element_rect(fill = "white"),
    axis.line = element_line(color = "black"),
    legend.position = "none")
```

#R statistics:

```
model <- lm(`Number of Fatalities` ~ `Suspected Impairment`, data = project)
rsquared <- summary(model)$r.squared
adjusted_rsquared <- summary(model)$adj.r.squared
standard_error <- summary(model)$sigma
cat("R-squared value:", rsquared, "\n")
cat("Adjusted R-squared:", adjusted_rsquared, "\n")
cat("Standard Error:", standard_error, "\n")
```



Hypothesis 5: Proportionally, more traffic fatalities occur during the weekend than during weekdays in Austin, Texas.
R-squared value: 0.009688 (very low correlation)
Adjusted R-squared: -0.0035657
Standard Error: 0.256429

#number of fatalities by weekday/weekend

```r
project$Number.of.Fatalities <- as.numeric(project$Number.of.Fatalities)

project <- project %>%
  mutate(Weekday = ifelse(Day %in% c("sat", "sun"), "Weekend", "Weekday"))
```

# Aggregate the data to get the count of fatalities for each weekday/weekend

```r
agg_data <- project %>%
  group_by(Weekday) %>%
  summarise(Fatalities = sum(Number.of.Fatalities))
```

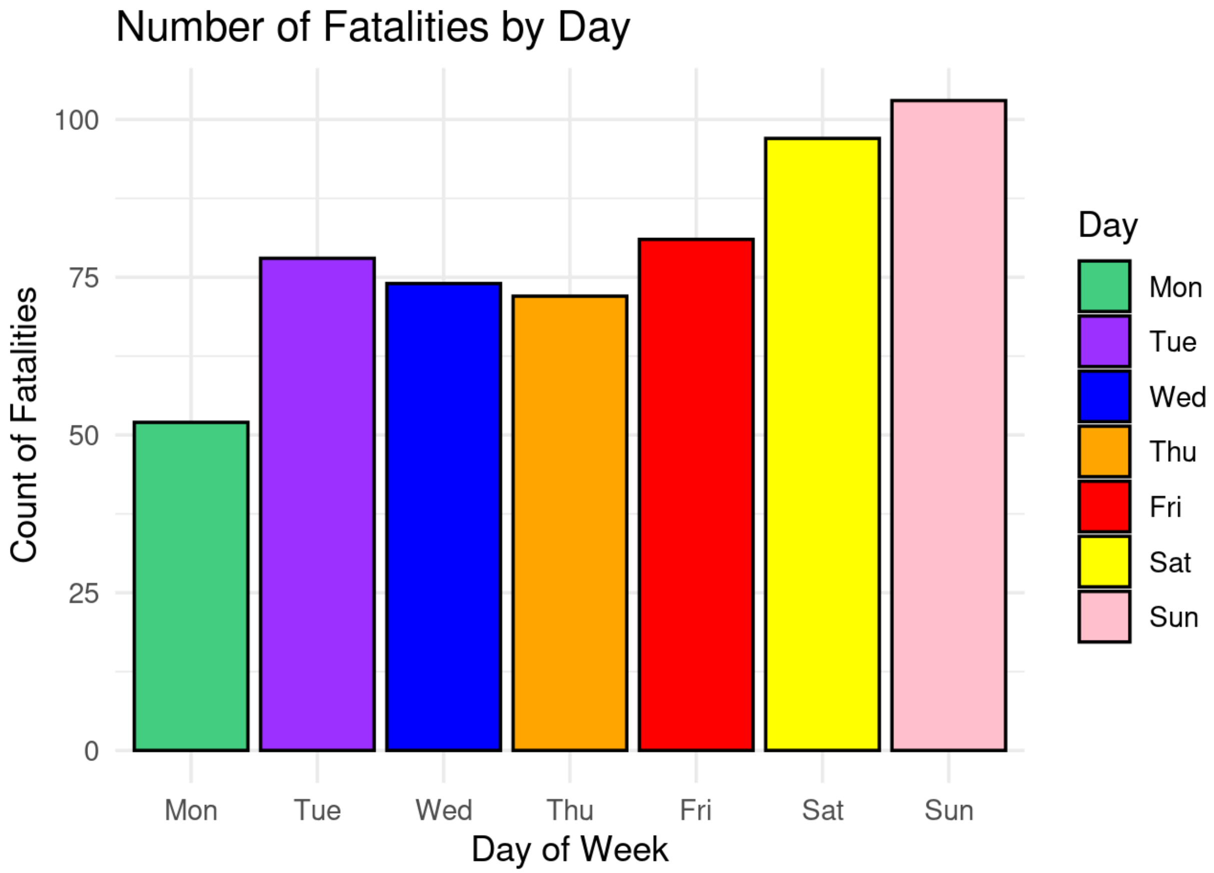
# Create a bar plot

```r
ggplot(agg_data, aes(x = Weekday, y = Fatalities, fill = Weekday)) +
  geom_bar(stat = "identity", position = "dodge", color = "black") +
  labs(title = "Number of Fatalities by Weekday/Weekend", x = "Day of Week", y =
"Count of Fatalities") +
  scale_fill_manual(values = c("Weekday" = "seagreen3", "Weekend" = "purple1")) +
  theme_minimal()
```

#R statistics:

```r
model <- lm(`Number of Fatalities` ~ `Day`, data = project)
rsquared <- summary(model)$r.squared
adjusted_rsquared <- summary(model)$adj.r.squared
standard_error <- summary(model)$sigma
cat("R-squared value:", rsquared, "\n")
cat("Adjusted R-squared:", adjusted_rsquared, "\n")
cat("Standard Error:", standard_error, "\n")
```

Hypothesis 5.5: More traffic fatalities occur on the weekend

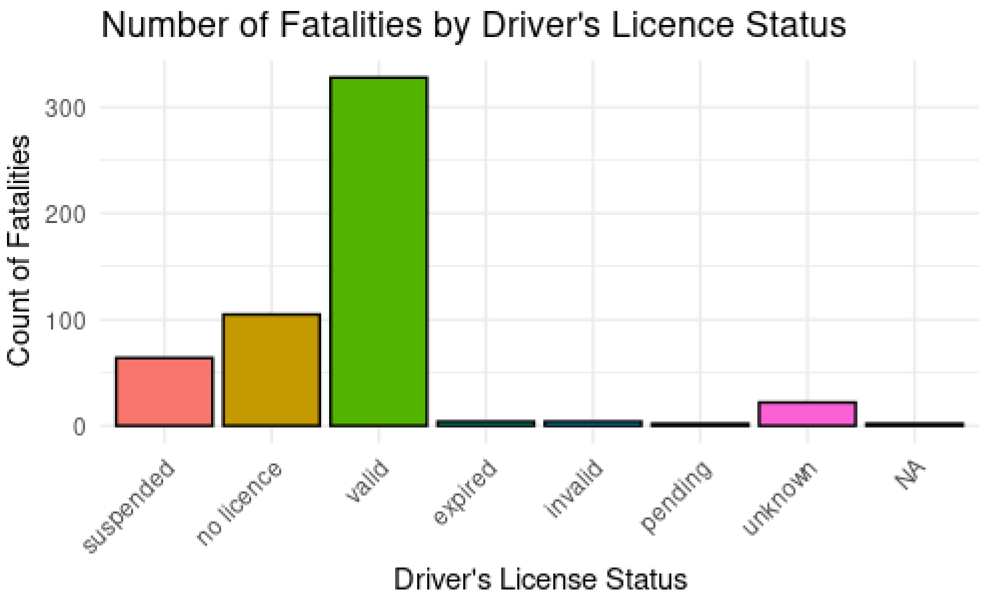


#R statistics:

```
project$Day <- factor(project$Day, levels = c("Mon", "Tue", "Wed", "Thu", "Fri", "Sat",
"Sun"))

project$Number.of.Fatalities <- as.numeric(project$Number.of.Fatalities)

agg_data <- project %>%
  group_by(Day) %>%
  summarise(Fatalities = sum(Number.of.Fatalities))

ggplot(agg_data, aes(x = Day, y = Fatalities, fill = Day)) +
  geom_bar(stat = "identity", position = "dodge", color = "black") +
  labs(title = "Number of Fatalities by Day", x = "Day of Week", y = "Count of Fatalities") +
  theme_minimal() +
  scale_fill_manual(values = c("Mon" = "seagreen3", "Tue" = "purple1", "Wed" = "blue",
                               "Thu" = "orange", "Fri" = "red", "Sat" = "yellow", "Sun" = "pink"))
```

Linear Regression Model by Day:

```
model <- lm(Number.of.Fatalities ~ Day, data = project)
summary(model)
```



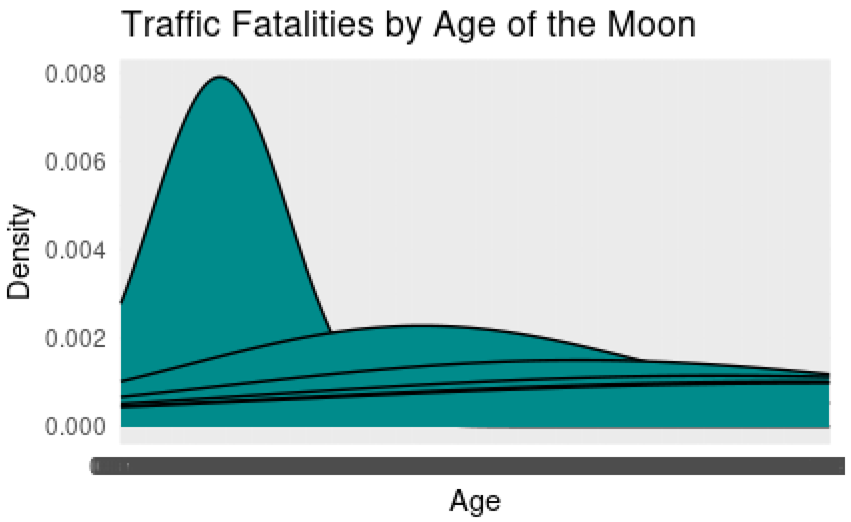
Hypothesis 6: Those with expired or suspended licenses are less likely to be involved in traffic fatalities.

R-squared value: 0.0677567 (low correlation)
Adjusted R-squared: 0.007612065
Standard Error: 0.2554623

```
ggplot(project, aes(x = DL.Status.incident, fill = DL.Status.incident)) +
  geom_bar(stat = "count", position = "dodge", color = "black") +
  labs(title = "Number of Fatalities by Driver's Licence Status",
           x = "Driver's License Status",
            y = "Count of Fatalities") +
   theme_minimal() +
  theme(axis.text.x = element_text(angle = 45, hjust = 1),
            legend.position = "none")
```

#R statistics:

```
model <- lm(`Number of Fatalities` ~ `DL Status incident`, data = project)
rsquared <- summary(model)$r.squared
adjusted_rsquared <- summary(model)$adj.r.squared
standard_error <- summary(model)$sigma
cat("R-squared value:", rsquared, "\n")
cat("Adjusted R-squared:", adjusted_rsquared, "\n")
cat("Standard Error:", standard_error, "\n")
```



# Create a density plot

```
ggplot(project, aes(x = Age)) +
  geom_density(fill = "cyan4", color = "black") +
  labs(title = "Traffic Fatalities by Age of the Moon",
       x = "Age",
       y = "Density") +
  theme_minimal()
```

Modeling

The task chosen was to predict whether or not a specific moon phase data affected driver behavior to cause traffic fatalities.

To complete this task, a classification model is required. A Random Forest model would suffice given tasks where a single Decision Tree may be too sensitive to the specifics of the training data. In this way, contrasts in feature strengths will show greater and give decipherable information.

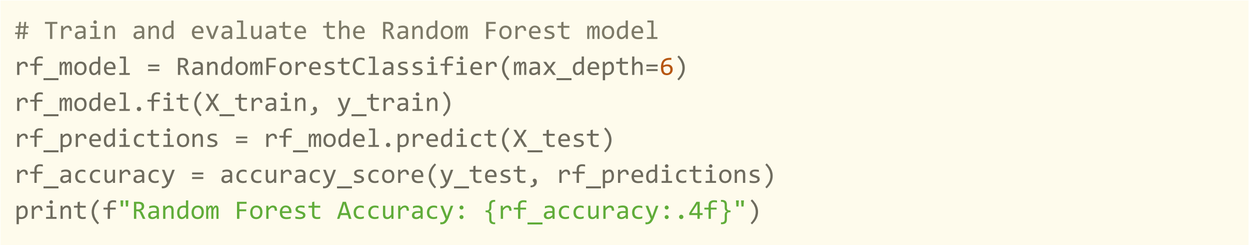
Preprocessing the data required dropping class unrelated features such as concrete observations of the moon, “street names”, “extraneous date information”, and “case numbers”. It also involved mutating and grouping categorical values into numerical ones using a label encoder.

```
#dropping unnecessary features
df = df.drop(['Fatal.Crash.Number', 'Unnamed: 0', 'Case.Number', 'Location',
'DateTime', 'Date'], axis=1)

#encoding categorical variables
from sklearn.preprocessing import LabelEncoder
label_encoder = LabelEncoder()
df['Type'] = label_encoder.fit_transform(df['Type'])
df['Month'] = label_encoder.fit_transform(df['Month'])
df['Day'] = label_encoder.fit_transform(df['Day'])
df['Killed.driver.pass'] = label_encoder.fit_transform(df['Killed.driver.pass'])
df['Speeding'] = label_encoder.fit_transform(df['Speeding'])
df['Ran.Red.Light.or.Stop.Sign'] =
label_encoder.fit_transform(df['Ran.Red.Light.or.Stop.Sign'])
df['DL.Status.incident'] = label_encoder.fit_transform(df['DL.Status.incident'])
df['Suspected.Impairment'] =
label_encoder.fit_transform(df['Suspected.Impairment'])
df['Type.of.road'] = label_encoder.fit_transform(df['Type.of.road'])
df['MoonPhaseCat'] = label_encoder.fit_transform(df['MoonPhaseCat'])
```

Then, assign “Moon Phase Category” as the target variable and all other columns as features; and use a random forest model to first split the data into 80% training data and 20% test data.

```
# Define the target variable and features
target_variable = 'MoonPhaseCat'
features = ['Type', 'Number.of.Fatalities', 'Month', 'Day', 'Hour',
       'Killed.driver.pass', 'Ran.Red.Light.or.Stop.Sign',
       'DL.Status.incident', 'Suspected.Impairment', 'Type.of.road',
       'Phase', 'Age', 'Diam', 'Dist', 'RA', 'Dec', 'Slon',
       'Slat', 'Elon', 'Elat', 'AxisA']

# Split the data into training and testing sets
X = df[features]
y = df[target_variable]
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2,
random_state=42)
```

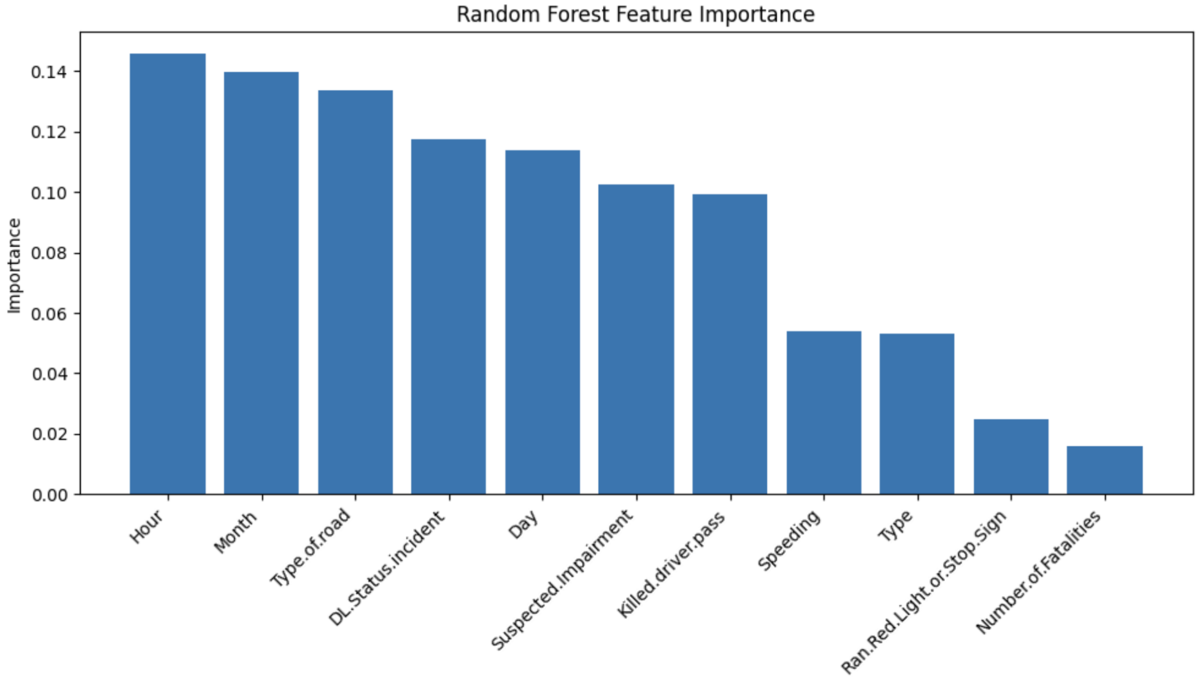
Using the random forest model, we train and evaluate the accuracy of the model.



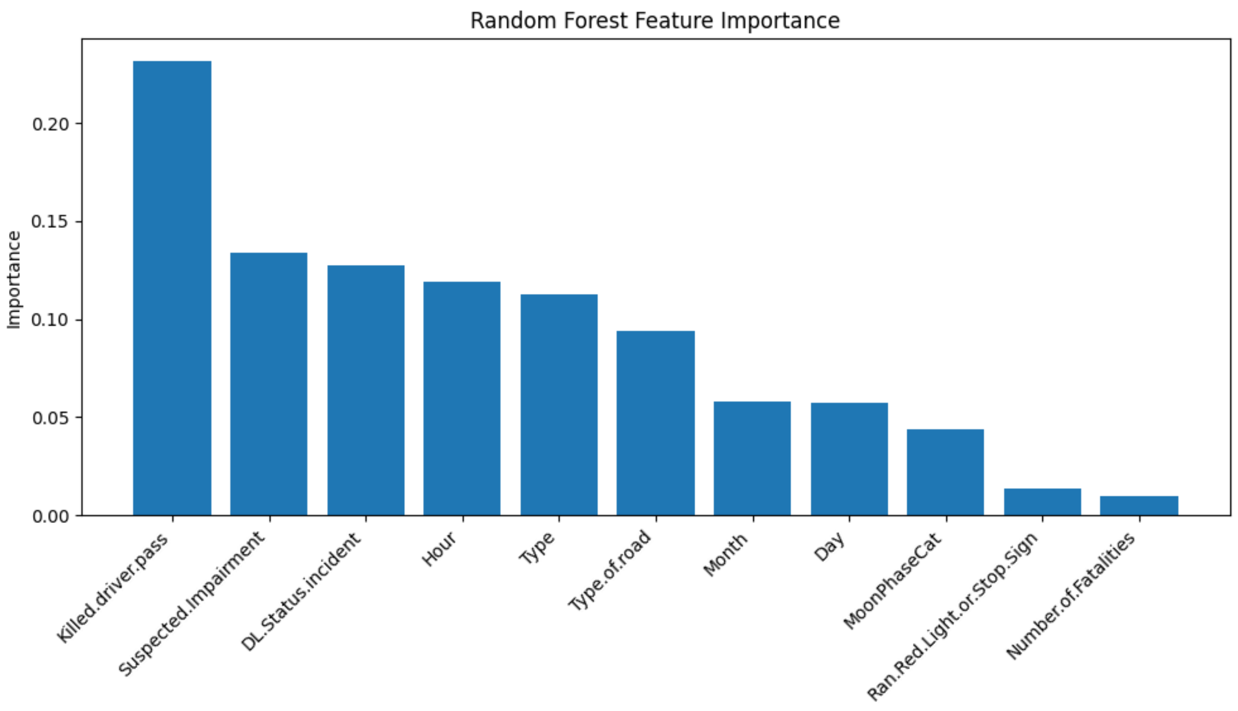
```python
# Train and evaluate the Random Forest model
rf_model = RandomForestClassifier(max_depth=6)
rf_model.fit(X_train, y_train)
rf_predictions = rf_model.predict(X_test)
rf_accuracy = accuracy_score(y_test, rf_predictions)
print(f"Random Forest Accuracy: {rf_accuracy:.4f}")
```

Random Forest Accuracy: 0.1682
The accuracy of the model results turns out to be a dismal 0.1682. The model has difficulty in modeling the feature data with the correct moon classification, showing very little correlation between the training data features and driver behavior.

Random Forest Classification Model Feature Weights



Running a model with speeding as the target variable further explains the lack of correlation between traffic fatalities and moon phases. Analysis of features importance also gives insight to strong causal and outcome predictors of reckless behavior in traffic fatalities.



<u>Discussion</u>

- Model's performance
  -
- Findings
  - The Random Forest Classification Model shows that the most important features are the Time of Day (Hour), Month and Road Type.
    - Further analysis of the effects of these explanatory variables on traffic fatalities may allow policymakers to better allocate funds and resources in a bid to promote policies and legislation to reduce the number of traffic fatalities in Austin.
  - The graph on number of fatalities by hour, shows that there are significantly less traffic fatalities between hour 7 and hour 19. This represents the time period of 7am-7pm, which is roughly when the sun rises and sets, indicating that more traffic fatalities occur at nighttime, when there is less light.
    - City legislators could potentially use this data to reevaluate road safety and policies in a bid to reduce traffic fatalities. For example, they could conduct further testing and evaluate the level of light available to drivers at nighttime in high-traffic areas.
    - Moreover, another peak in the data occurs at around Hour 2 or 2am. In Austin, this is also when many of the bars/clubs close downtown.

Potentially this surge and high concentration of people trying to get home is a contributing factor to the fatalities occurring at this time. Legislators could look at increasing police presence in high traffic volume areas or alternatives such as shuttle buses from downtown to high density neighborhoods to try and reduce traffic fatalities.

- Limitations
    - The disparity between fatalities by differing driver’s license status is more likely down to the high proportion of driver’s with valid licenses on the road rather than driver’s with valid licenses causing more traffic fatalities. We cannot draw conclusions from the graph we created since it does not account for these proportions.
    - The analysis between traffic fatalities and road type highlights a potential issue when working with qualitative variables, since the distinction between different road types is more difficult to discern, it is also harder to distinguish conclusions from the data.
    - It seems that there are fairly similar amounts of fatalities between driver’s with suspected impairment and driver’s with no suspected impairment but this also fails to account for the proportion of drivers that are impaired or not impaired. It is likely that if proportion were factored in, despite the absolute number of fatalities being fairly similar, that the proportion of impaired drivers in fatal accidents is higher than the proportion of non-impaired drivers.

Ethics

- Unintended Consequences - (Reflect on Product)
    - The dip in traffic fatalities during first and last quarters does not reflect the overall dip in traffic fatalities, rather it reflects the data removed during preprocessing as a result of non-matching occurrences between corresponding moon phases and data-times recorded.
    - It's crucial to consider unintended consequences that may arise from our analysis of traffic fatalities and moon phases. One unintended consequence could be the misinterpretation of correlation as causation. One may misinterpret the findings as the moon has definite effects on traffic fatalities and that fatal traffic events are more likely to occur when the moon is in waxing gibbous, or less likely during first and last quarters. This oversimplification could lead to misconceptions and a dismissal of other relevant factors influencing traffic accidents.

- People Affected - (Anticipate People)
    - The unintended consequences may cause changes in public perception and behavior. If media outlets or individuals sensationalize the findings, there is a risk

that some Austin drivers alter their behavior based on the perceived influence of the moon. For example, drivers might become overly cautious during certain moon phases or overly confident during others, leading to changes in driving patterns that could impact road safety.

- Continuous Improvement - (Act on Process)
    - To mitigate potential misconstruction, our data needs to clearly communicate the lack of a significant correlation between moon phases and traffic fatalities. Descriptions should provide context on the limitations of the analysis and the importance of considering multiple factors influencing road safety; acknowledge uncertainties and the need for further research to validate any observed patterns.

Conclusion

Overall, in our study to find factors that correlate to higher traffic fatalities in Austin, Texas, we hypothesized that a waxing gibbous would have more traffic fatalities, high use roadways would have more traffic fatalities, daylight hours would have less traffic fatalities, drivers suspected to be impaired would have more traffic fatalities, weekend would have more traffic fatalities, and those with expired/suspended licenses are less likely to be involved in traffic fatalities. We compiled the data to test these relationships from the Traffic Fatality Dataset published by the Austin government from 2013 through 2019 and combined it with data of the recorded moon phases during these years recorded by Nasa. After cleaning the data we used a Random Forest Classification Model to find the most significant variables to be time of day, month, and road type. Further analyzing the data by producing graphs showed that there is a significantly lower amount of traffic fatalities from 7am to 7pm indicating that more traffic fatalities occur at night, specifically peaking at 2am and 9pm. This finding indicates that presence of less light does impact the amount of traffic fatalities that occur. Graphs of traffic fatalities during the days of the week showed higher rates during Saturday and Sunday, though these differences were proven insignificant through R squared analysis. In concern of our other hypotheses, we found that the extremely unequal number of drivers with valid driver's licenses in comparison to those without made the graphs inconclusive on the overall effect. Additionally, the qualitative variables in the different road types made the data harder to analyze due to too much variation in the possible options and the possible inaccuracies of suspected impairment rather than actual impairment made the data unreliable. The cleaning of the moon phase data also resulted in a large loss of data points that made our analyses of it inaccurate due to missing data points and non-matching occurrences. Despite these difficulties, we believe that the information from our study can be used to urge policymakers to reevaluate road safety during times of low light availability and work towards artificial light solutions that more effectively prevent traffic fatalities during these times. Further testing of this research could be performed through analyzing the quality of the artificial lighting installed on roads to see what improvements need to be made to increase visibility and lessen traffic fatalities at night.

Acknowledgement

Name	Contributions	Contribution Percentage
Lauren Platz	Combining data sets, preparing google slides for presentation, report introduction, bibliography, data description, visualizations and hypotheses	100%
Kyle Jones	Finding data sets, data cleaning, combining data sets	100%
Ashley Schluter	R statistics code and analysis, presentation, conclusion	100%
Donne Su	Ethics evaluation, modeling, github report, cleaning data, description, website	100%
Kevin Zhou	Evaluation and R analysis of results and findings, implications, presentation content.	100%
Trina Nguyen	Introduction, assumptions and justifications of project, transformed slides to Canva presentation, presentation aesthetics	100%

Bibliography

City of Austin, Texas - data.austintexas.gov. “2013 APD Traffic Fatalities: Open Data: City of Austin Texas.” *Data.AustinTexas.Gov - The Official City of Austin Open Data Portal*, 14 Mar. 2018, data.austintexas.gov/Public-Safety/2013-APD-Traffic-Fatalities/vggi-9ddh.

City of Austin, Texas - data.austintexas.gov. “2014 APD Traffic Fatalities: Open Data: City of Austin Texas.” *Data.AustinTexas.Gov - The Official City of Austin Open Data Portal*, 14

Mar. 2018, data.austintexas.gov/Public-Safety/2014-APD-Traffic-Fatalities/gm9p-snyb.

City of Austin, Texas - data.austintexas.gov. “2015 APD Traffic Fatalities: Open Data: City of Austin Texas.” *Data.AustinTexas.Gov - The Official City of Austin Open Data Portal*, 14 Mar. 2018, data.austintexas.gov/Public-Safety/2015-APD-Traffic-Fatalities/p658-umsa.

City of Austin, Texas - data.austintexas.gov. “2016 APD Traffic Fatalities: Open Data: City of Austin Texas.” *Data.AustinTexas.Gov - The Official City of Austin Open Data Portal*, 14 Mar. 2018, data.austintexas.gov/Public-Safety/2016-APD-Traffic-Fatalities/tiqb-wv3c.

City of Austin, Texas - data.austintexas.gov. “2017 APD Traffic Fatalities: Open Data: City of Austin Texas.” *Data.AustinTexas.Gov - The Official City of Austin Open Data Portal*, 25 Sept. 2018, data.austintexas.gov/Public-Safety/2017-APD-Traffic-Fatalities/ijds-pcyq.

City of Austin, Texas - data.austintexas.gov. “2018 APD Traffic Fatality Data 021219: Open Data: City of Austin Texas.” *Data.AustinTexas.Gov - The Official City of Austin Open Data Portal*, 12 Feb. 2019, data.austintexas.gov/Public-Safety/2018-APD-Traffic-Fatality-Data-021219/9jd4-zjmx.

City of Austin, Texas - data.austintexas.gov. “2019 APD Traffic Fatality Data: Open Data: City of Austin Texas.” *Data.AustinTexas.Gov - The Official City of Austin Open Data Portal*, 24 Apr. 2020, data.austintexas.gov/Public-Safety/2019-APD-Traffic-Fatality-Data/egpd-hqdi.

Wright, Ernie. “NASA Scientific Visualization Studio.” *NASA*, NASA, 24 Oct. 2023, svs.gsfc.nasa.gov/4442.

Code:

```
ggplot(project, aes(x = MoonPhaseCat)) +
```

```r
  geom_bar(fill = "mediumslateblue") +
  labs(title = "Number of Fatalities by Moon Phase Category", x = "Moon Phase
Category", y = "Count of Fatalities") +
  theme(axis.text.x = element_text(angle = 45, hjust = 1))

project <- project %>%
  mutate_all(tolower)
project <- project %>%
  mutate(Type.of.road = ifelse(str_detect(Type.of.road, "35"), "IH-35",
Type.of.road))
project <- project %>%
  mutate(Type.of.road = ifelse(str_detect(Type.of.road, "high speed"), "high speed
roadway", Type.of.road))

project <- project %>%
  mutate(Type.of.road = ifelse(str_detect(Type.of.road, "use"), "high use roadway",
Type.of.road))
project <- project %>%
  mutate(Type.of.road = ifelse(str_detect(Type.of.road, "highspeed"), "high speed
roadway", Type.of.road))

#number of fatalities by road type
project %>%
  group_by(Type.of.road) %>%
  summarise(FatalitiesCount = n()) %>%
  ggplot(aes(x = reorder(Type.of.road, FatalitiesCount), y = FatalitiesCount)) +
  geom_bar(fill = "aquamarine3", stat = "identity") +
  labs(title = "Number of Fatalities by Road Type", x = "Road Type", y = "Count of
Fatalities") +
  theme(axis.text.x = element_text(angle = 45, hjust = 1))

#number of fatalities by hour
project$Hour <- as.numeric(project$Hour)

ggplot(project, aes(x = Hour)) +
  geom_line(stat = "count", color = "darkslateblue", size = 1.5) +
  labs(title = "Number of Fatalities by Hour", x = "Hour", y = "Count of
Fatalities") +
  theme_minimal() +
  theme(
    panel.grid.major = element_blank(),
    panel.grid.minor = element_blank(),
    panel.background = element_rect(fill = "white"),
    axis.line = element_line(color = "black"),
    axis.text = element_text(size = 10),
    axis.title = element_text(size = 12, face = "bold"),
    plot.title = element_text(size = 16, face = "bold", hjust = 0.5),
    legend.position = "none"  # Remove legend if not needed)
project <- project %>%
  mutate(Suspected.Impairment = ifelse(str_detect(Suspected.Impairment, "bic"),
"bicyclist", Suspected.Impairment))
project <- project %>%
  mutate(Suspected.Impairment = ifelse(str_detect(Suspected.Impairment, "both"),
"both", Suspected.Impairment))
project <- project %>%
  mutate(Suspected.Impairment = ifelse(str_detect(Suspected.Impairment, "driver"),
"driver", Suspected.Impairment))
project <- project %>%
  mutate(Suspected.Impairment = ifelse(str_detect(Suspected.Impairment, "unk"),
"unknown", Suspected.Impairment))
project <- project %>%
  mutate(Suspected.Impairment = ifelse(str_detect(Suspected.Impairment, "ped"),
"pedestrian", Suspected.Impairment))
project <- project %>%
  mutate(Suspected.Impairment = ifelse(str_detect(Suspected.Impairment, "none"),
"none", Suspected.Impairment))

#number of fatalities by suspected impairment
project$Suspected.Impairment <- fct_reorder(project$Suspected.Impairment,
project$Number.of.Fatalities, .fun = sum, .desc = TRUE)

ggplot(project, aes(x = Suspected.Impairment)) +
  geom_bar(fill = "plum4", color = "black", size = 0.7, alpha = 0.8) +
  labs(title = "Number of Fatalities by Suspected Impairment", x = "Suspected
Impairment", y = "Count of Fatalities") +
  theme_minimal() +
  theme(
    axis.text.x = element_text(angle = 45, hjust = 1, vjust = 1),
    panel.grid.major = element_blank(),
    panel.grid.minor = element_blank(),
    panel.background = element_rect(fill = "white"),
    axis.line = element_line(color = "black"),
    legend.position = "none"))

#number of fatalities by weekday/weekend
project$Number.of.Fatalities <- as.numeric(project$Number.of.Fatalities)

project <- project %>%
  mutate(Weekday = ifelse(Day %in% c("sat", "sun"), "Weekend", "Weekday"))

# Aggregate the data to get the count of fatalities for each weekday/weekend
agg_data <- project %>%
  group_by(Weekday) %>%
  summarise(Fatalities = sum(Number.of.Fatalities))
```

```r
# Create a bar plot
ggplot(agg_data, aes(x = Weekday, y = Fatalities, fill = Weekday)) +
  geom_bar(stat = "identity", position = "dodge", color = "black") +
  labs(title = "Number of Fatalities by Weekday/Weekend", x = "Day of Week", y =
"Count of Fatalities") +
  scale_fill_manual(values = c("Weekday" = "seagreen3", "Weekend" = "purple1")) +
  theme_minimal()

project <- project %>%
  mutate(DL.Status.incident = ifelse(str_detect(DL.Status.incident, "suspended"),
"suspended", DL.Status.incident))
project <- project %>%
  mutate(DL.Status.incident = ifelse(str_detect(DL.Status.incident, "exp"),
"expired", DL.Status.incident))
project <- project %>%
  mutate(DL.Status.incident = ifelse(str_detect(DL.Status.incident, "no"), "no
licence", DL.Status.incident))
project <- project %>%
  mutate(DL.Status.incident = ifelse(str_detect(DL.Status.incident, "ok"), "valid",
DL.Status.incident))
project <- project %>%
  mutate(DL.Status.incident = ifelse(str_detect(DL.Status.incident, "unk"),
"unknown", DL.Status.incident))
project <- project %>%
  mutate(DL.Status.incident = ifelse(str_detect(DL.Status.incident, "invalid"),
"invalid", DL.Status.incident))
project <- project %>%
  mutate(DL.Status.incident = ifelse(str_detect(DL.Status.incident, "supsended"),
"suspended", DL.Status.incident))

#number of fatalities by driver's license status
ggplot(project, aes(x = DL.Status.incident, fill = DL.Status.incident)) +
  geom_bar(stat = "count", position = "dodge", color = "black") +
  labs(title = "Number of Fatalities by Driver's Licence Status",
           x = "Driver's License Status",
            y = "Count of Fatalities") +
   theme_minimal() +
  theme(axis.text.x = element_text(angle = 45, hjust = 1),
            legend.position = "none")

project <- project %>%
  mutate(Killed.driver.pass = ifelse(str_detect(Killed.driver.pass, "both"),
"both", Killed.driver.pass))
project <- project %>%
  mutate(Killed.driver.pass = ifelse(str_detect(Killed.driver.pass, "motorcycle"),
"motorcyclist", Killed.driver.pass))
```

```r
project <- project %>%
  mutate(Killed.driver.pass = ifelse(str_detect(Killed.driver.pass, "ped"),
"pedestrian", Killed.driver.pass))
project <- project %>%
  mutate(Killed.driver.pass = ifelse(str_detect(Killed.driver.pass, "driver &
passenger"), "driver and passenger", Killed.driver.pass))
project <- project %>%
  mutate(Killed.driver.pass = ifelse(str_detect(Killed.driver.pass, "mc driver"),
"motorcyclist", Killed.driver.pass))
project <- project %>%
  mutate(Killed.driver.pass = ifelse(str_detect(Killed.driver.pass, "driver
(other)"), "other driver", Killed.driver.pass))
project <- project %>%
  mutate(Killed.driver.pass = ifelse(str_detect(Killed.driver.pass, ""), "both",
Killed.driver.pass))

# Create a line graph
ggplot(project, aes(x = Dist, y = ..count.., group = 1)) +
  geom_line(stat = "count", color = "skyblue") +
  labs(title = "Number of Fatalities by Phase",
       x = "Phase",
       y = "Count of Fatalities") +
  theme_minimal()
```