STAT 187 Final Project

Connor Constantino, Delaney Woods, Nicholas Gibson

5/11/2022

```
knitr::opts_chunk$set(echo = TRUE,
                      fig.width=8,
                      fig.height=5)

pacman::p_load(tidyverse, skimr, stringr, lubridate,
               socviz, grid, usmap, maps, statebins, viridis, leaflet, rgdal,
               broom, maptools, rgeos,
               caret, class, caTools, rpart, rpart.plot)

theme_map <- function(base_size=9, base_family="") {
    require(grid)
    theme_bw(base_size=base_size, base_family=base_family) %+replace%
        theme(axis.line=element_blank(),
              axis.text=element_blank(),
              axis.ticks=element_blank(),
              axis.title=element_blank(),
              panel.background=element_blank(),
              panel.border=element_blank(),
              panel.grid=element_blank(),
              panel.spacing=unit(0, "lines"),
              plot.background=element_blank(),
              legend.justification = c(0,0),
              legend.position = c(0,0)
              )
}
```

Intro

Driving is more than likely the most dangerous thing you will do within the next year. Every year millions of people die in car accidents in the US alone. What if we could determine the severity of a car crash before it happened and selectively restructure areas with a high level of danger.

In this project, we are looking at a dataset containing information on car crashes in Vermont from 2010-2022. The dataset we are using is from an observational study and contains an inherent bias towards Vermont drivers since the study was conducted in Vermont. This dataset contained a large amount of missing values that needed to be filtered out and many of the chr data types had to be converted into factors.

Our main goal is to investigate what factors affect the chances and outcomes of a car accident. For example we will investigate what conditions make crashes of different types of vehicles more likely and whether or not the accident causes injuries or fatalities. Using a decision tree on our dataset we will develop a model to be able to estimate whether or not a car crash will cause only property damage, injuries, or fatalities given certain conditions.

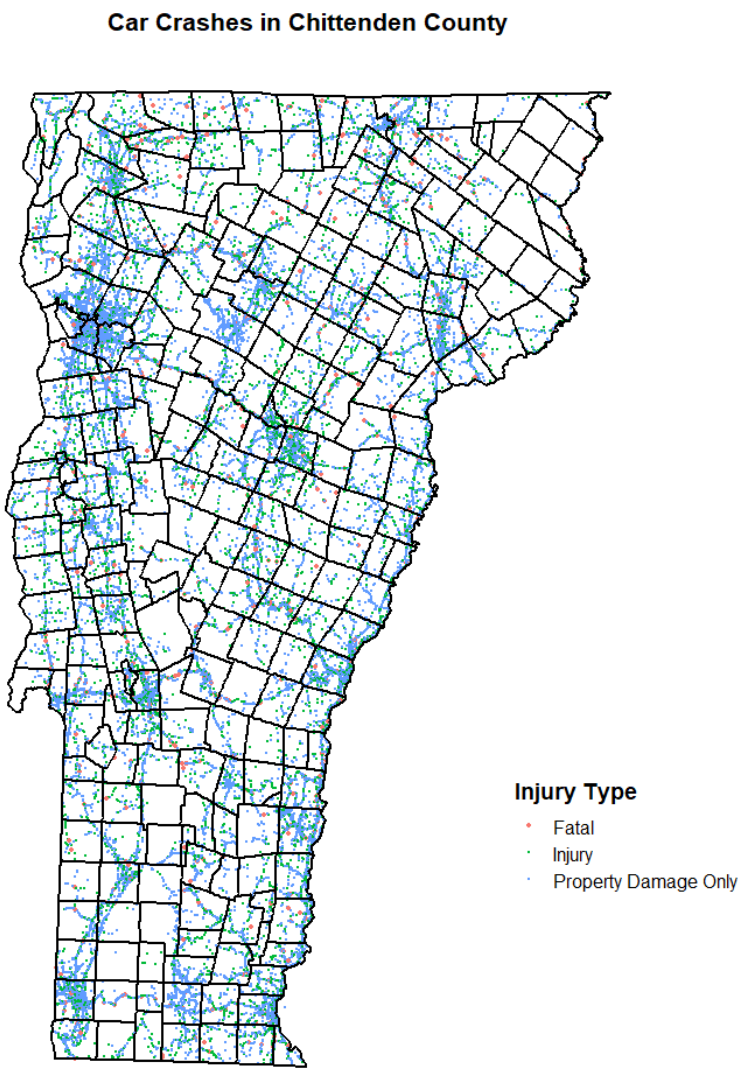
Data Visualization

```
# The full data set with all empty and unknown values stored as na
crash_data_na <- read.csv("VermontCrashData20102022.csv") %>%

  mutate_all(na_if, "") %>%

  mutate_all(na_if, "Unknown")

# The full data set with no conversion to na
crash_data_map1 = read.csv("VermontCrashData20102022.csv")
```

Crash Map

```
chittenden_shape <- readOGR(
dsn = "tl_2021_50_cousub",
layer = "tl_2021_50_cousub"
)

## OGR data source with driver: ESRI Shapefile
## Source: "C:\One Drive\OneDrive\[1] Documents\[3] Job
Applications\[Portfolio]\Car Crash Data\tl_2021_50_cousub", layer:
"tl_2021_50_cousub"
## with 255 features
## It has 18 fields
## Integer64 fields read as strings:  ALAND AWATER

chittenden_fortified <- broom::tidy(chittenden_shape, region = "NAME")

crash_data_map2 <- crash_data_map1 %>%

  na.omit() %>%

  mutate(InjuryType = as.factor(InjuryType)) %>%

  filter(InjuryType != "")

ggplot(data = chittenden_fortified,
       mapping = aes(x = long,
                     y = lat,
                     group = group)) +

  geom_point(data = crash_data_map2,
             mapping = aes(x = LONGITUDE,
```

```
                          y = LATITUDE,
           group = NA,
           color = InjuryType,
           size = InjuryType)) +

labs(title = 'Car Crashes in Chittenden County',
     color = 'Injury Type',
     size = 'Injury Type') +

geom_polygon(fill = "transparent", color = "transparent") +

coord_map() +

theme_map() +

geom_path(color = "black", size = 1) +

theme(legend.position = c(0.8, 0.2),
      legend.title = element_text(size = 14, face = 'bold'),
      legend.text = element_text(size = 12),
      plot.title = element_text(face = 'bold', hjust = 0.5, size = 16)) +

scale_size_manual(name = "Injury Type", values = c(1, .1, .1))
```



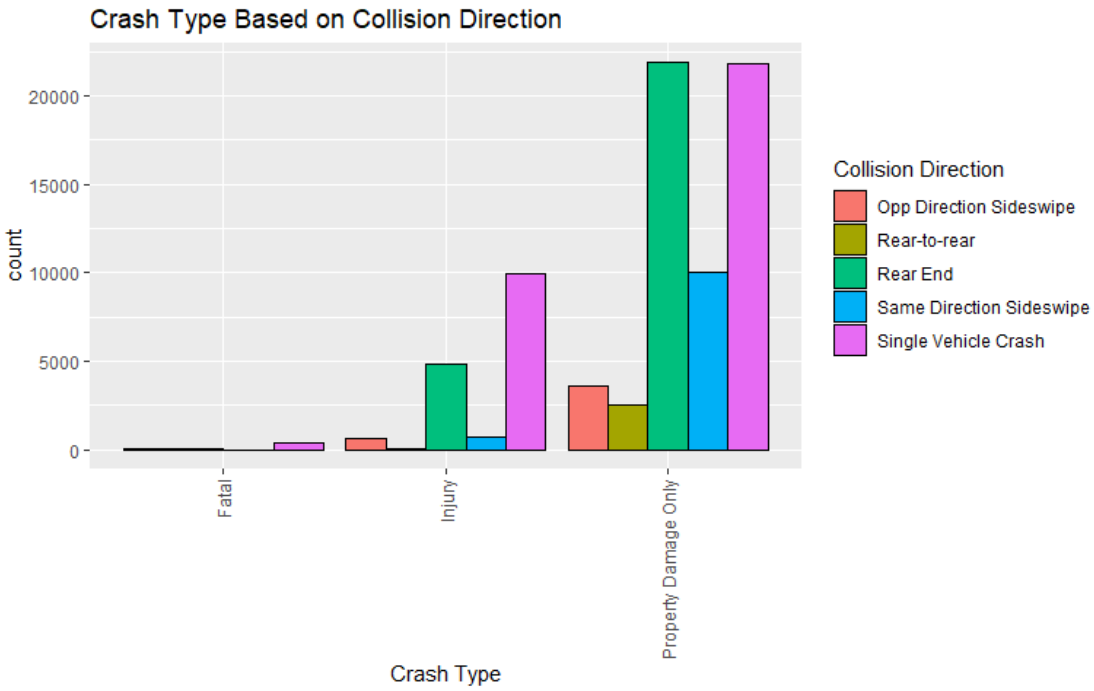
Explanation

This graph shows that the highest density of car crashes in Vermont happen along the highways and seem to be clustered around cities. This could be due to a higher population density leading to more cars on the roads in those areas. However, it could also be attributed to poor road systems throughout Vermont cities like Burlington and Montpelier both of which have some of the highest car crash rates in Vermont. We also see a relatively random distribution of fatal car crashes throughout this graph so location doesn't seem to have an affect on the fatality of a car accident.

Collisions

```
crash_data_dir <- crash_data_na %>%

  select(c(InjuryType, DirOfCollision)) %>%

  na.omit() %>%
```

```
  filter(DirOfCollision %in% c("Rear End", "Single Vehicle Crash",
                               "Head on", "Opp Direction Sideswipe",
                               "Same Direction Sideswipe", "Rear-to-rear"))
%>%

  filter(InjuryType %in% c("Fatal", "Injury", "Property Damage Only") )

crash_data_dir %>%

  group_by(InjuryType, DirOfCollision) %>%

  summarise(count = n()) %>%

  ungroup() %>%

  ggplot(mapping = aes(x = InjuryType,
                       y = count,
                       fill = DirOfCollision)) +

  geom_bar(stat = "identity",
           position = "dodge",
           color = "black") +

  labs(x = "Crash Type",
       fill = "Collision Direction",
       title = "Crash Type Based on Collision Direction") +

  theme(axis.text.x = element_text(angle=90, vjust=.5, hjust=1))

## `summarise()` has grouped output by 'InjuryType'. You can override using
the
## `.groups` argument.
```



Explanation

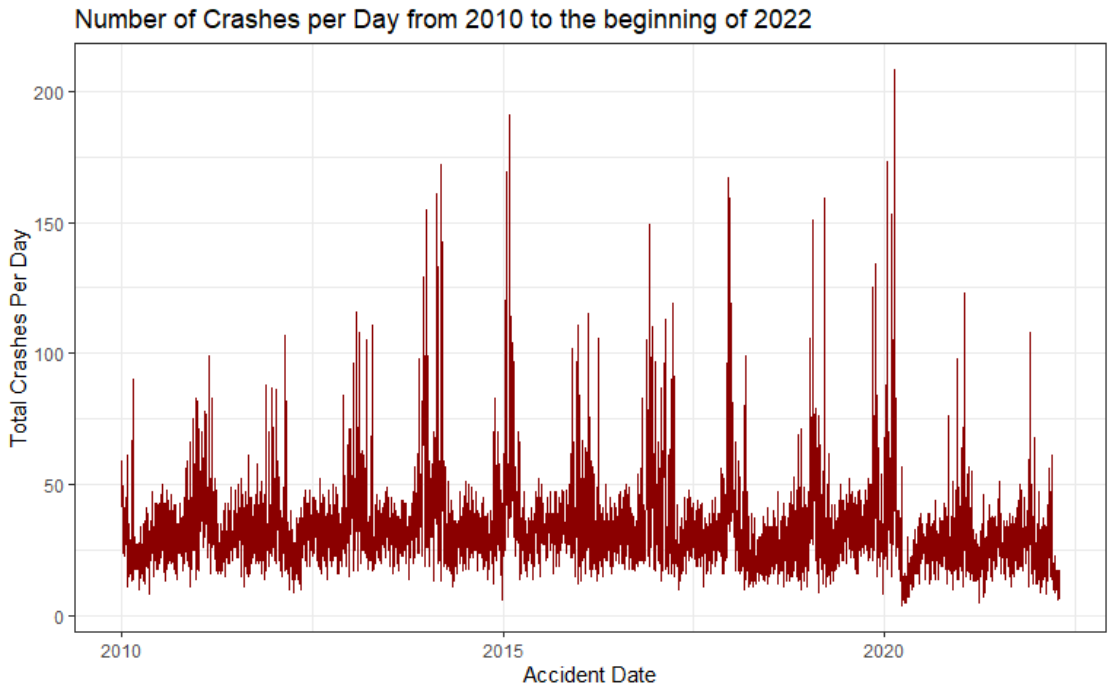
This bar chart reveals first that the most common type of crash type is property damage only. This graph also shows how rare a fatal crash type is and that this crash type only occurs with a single vehicle crash in our data set. In the property damage only, rear end and single vehicle crash are almost tied and are the most common collision direction by far. For injury, single vehicle crash is the most common collision direction and rear end is smaller.

It makes sense that the most common type of crash type is property damage only because this is the least severe type of crash. It also makes sense that rear ends and single vehicle crash have the highest counts because these are the more common accident.

Accidents per Day

```
# Extract the date and time from the ACCIDENTDATE column use regex,
# then use lubridate to format the dates
crash_data_datetime <- crash_data_na %>%

  select(ACCIDENTDATE) %>%

  na.omit() %>%

  mutate(AccidentDate = str_extract(string = ACCIDENTDATE,
         pattern = "(\\d\\d|\\d)/(\\d\\d|\\d)/(\\d\\d\\d\\d)")) %>%

  mutate(AccidentDate = mdy(AccidentDate)) %>%

  mutate(AccidentTime = str_extract(string = ACCIDENTDATE,
                                    pattern = "
```

```
(\\d\\d|\\d):(\\d\\d|\\d):(\\d\\d|\\d)")) %>%

  mutate(AccidentTime = hms(AccidentTime)) %>%

  select(-ACCIDENTDATE)

crash_data_datetime %>%

  group_by(AccidentDate) %>%

  summarise(Total = n()) %>%

  ungroup() %>%

  ggplot(mapping = aes(x = AccidentDate,
                       y = Total)) +

  geom_line(color = "darkred") +

  labs(y = "Total Crashes Per Day",
       x = "Accident Date",
       title = "Number of Crashes per Day from 2010 to the beginning of
2022") +

  theme_bw()
```



Explanation

This line graph shows the total number of crashes per day in the whole state of Vermont from 2010 to 2020. There appears to be a cyclical pattern where this a high point about every year. The very highest point appears to be in the year 2020, which is then followed by a pretty dramatic drop to one of the lowest points. This would make sense since lock down for COVID-19 started in March of 2020 and less people were out on the road.

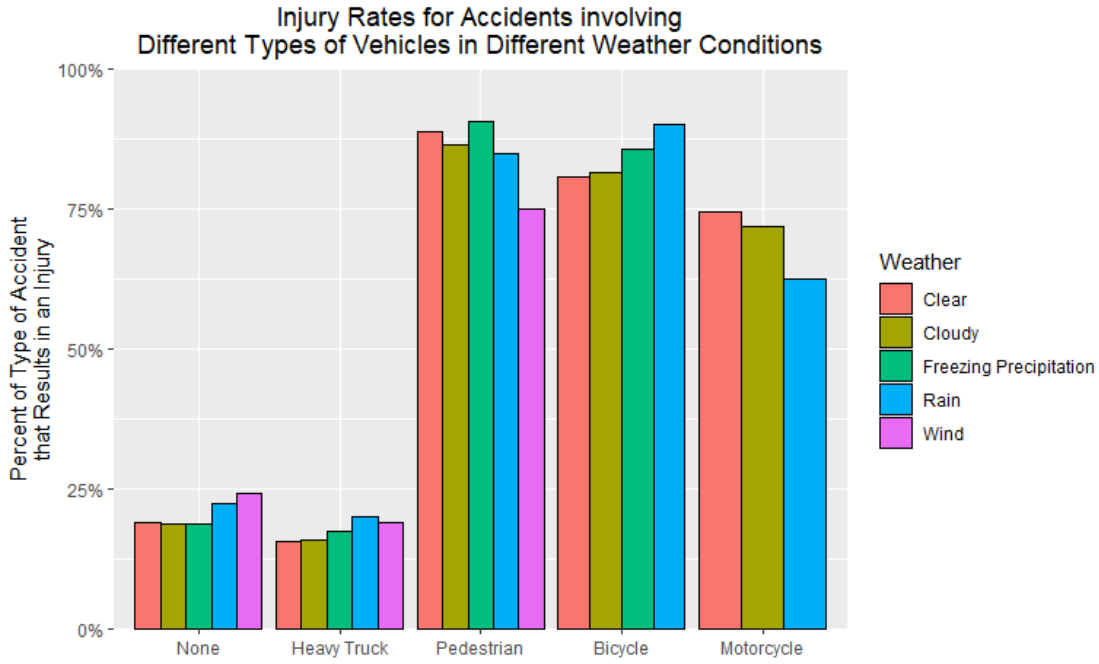
Looking at the 2015 point and the 2020 point, it seems like the high points are towards the beginning of the year, maybe February or March. This suggests that weather, including snow or ice, could impact the number of crashes.

Effect of Vehicle Type and Weather on Injury and Fatality Rates

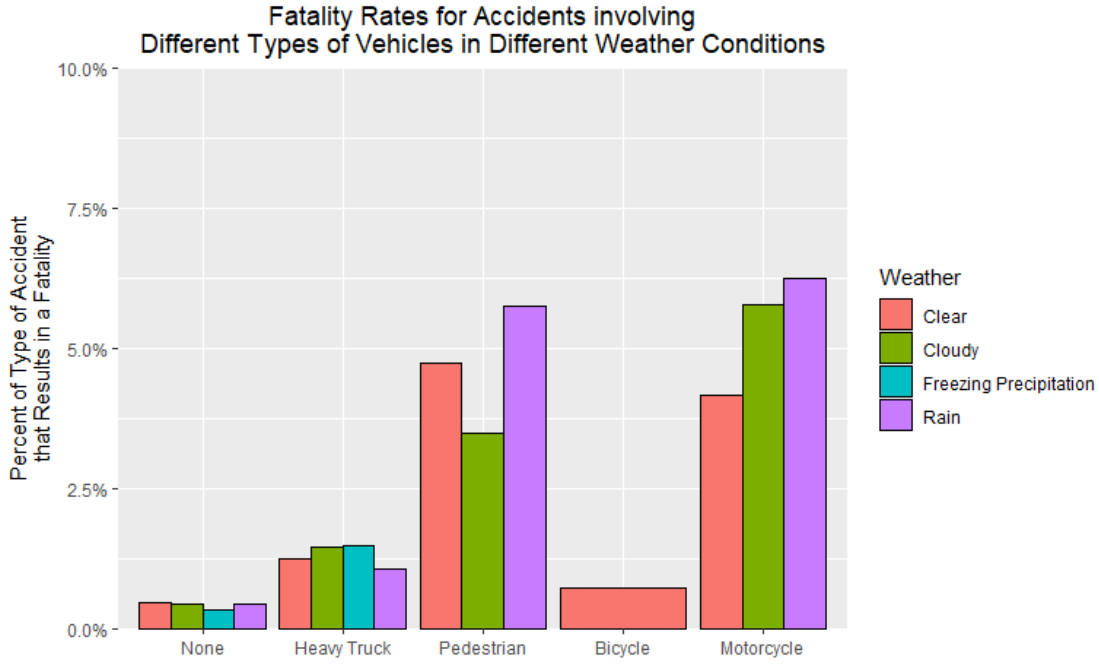
```
# Select only the columns we will use
crash_data_involving <- crash_data_na %>%

  dplyr::select(ACCIDENTDATE, Weather, DayNight, Involving,
         Impairment, Animal, InjuryType) %>%

  na.omit() %>%

  mutate(Involving = factor(Involving,
                            levels = c("None", "Heavy Truck", "Pedestrian",
                                       "Bicycle", "Motorcycle")))

# Total crashes for each group of involving and weather
totals <- crash_data_involving %>%

  select(Involving, Weather) %>%

  group_by(Involving, Weather) %>%

  summarise(TypeTotal = n())

## `summarise()` has grouped output by 'Involving'. You can override using
the
## `.groups` argument.

# Get the totals of each combination of involving, weather, and injury type
injury_data <- crash_data_involving %>%

  group_by(Involving, Weather, InjuryType) %>%

  summarise(Num = n()) %>%

  ungroup()
```

```
## `summarise()` has grouped output by 'Involving', 'Weather'. You can
override
## using the `.groups` argument.

# Add in the group totals
injury_data <- left_join(x = injury_data,
                         y = totals)

## Joining, by = c("Involving", "Weather")

# Calculate the rate
injury_data <- injury_data %>%

  mutate(Rate = Num / TypeTotal)


# Separate the injury and fatal rates into different columns
injury_rate <- injury_data %>%

  filter(InjuryType == "Injury") %>%

  select(Involving, Weather, Num, Rate) %>%

  rename(InjuryRate = Rate)

fatal_rate <- injury_data %>%

  filter(InjuryType == "Fatal") %>%

  select(Involving, Weather, Num, Rate) %>%

  rename(FatalRate = Rate)

injury_plot <-

  injury_rate %>% filter(Num >= 3) %>%

  ggplot(mapping = aes(x = Involving,
                       y = InjuryRate,
                       fill=Weather,
                       groups=1)) +

  geom_col(position = "dodge",
           color = "black") +

  scale_y_continuous(labels = scales::percent,
                     limits = c(0,1),
                     expand = expansion(mult = 0,
                                        add = c(0, 0))) +
```

```
  labs(title = "Injury Rates for Accidents involving\nDifferent Types of
Vehicles in Different Weather Conditions",
       x = "",
       y = "Percent of Type of Accident\nthat Results in an Injury") +

  theme(plot.title = element_text(hjust = 0.5))

fatal_plot <-

  fatal_rate %>% filter(Num >= 3) %>%

  ggplot(mapping = aes(x = Involving,
                       y = FatalRate,
                       fill=Weather,
                       groups=1)) +

  geom_col(position = "dodge",
           color = "black") +

  scale_y_continuous(labels = scales::percent,
                     limits = c(0, 0.1),
                     expand = expansion(mult = 0,
                                        add = c(0, 0))) +

  labs(title = "Fatality Rates for Accidents involving\nDifferent Types of
Vehicles in Different Weather Conditions",
       x = "",
       y = "Percent of Type of Accident\nthat Results in a Fatality") +

  theme(plot.title = element_text(hjust = 0.5))

injury_plot
```




```
fatal_plot
```



Explanation

The most obvious aspect of these graphs is how much more dangerous accidents are when they involve anyone not in a car. Accidents with pedestrians or any kind of bike have close to a 60% higher rate of injury compared to accidents involving cars. Interestingly, accidents involving heavy trucks have slightly lower injury rates than ones involving normal cars, but

this is presumably because they are much more likely to cause fatal accidents. Comparing bicycles and motorcycles is similar. You'd expect motorcycles to cause more injuries than bicycles due to their high speeds, but their lower injury rate is also likely due to a higher fatality rate.

When looking at the data for fatality rates, the most unexpected element is how low the fatality rate for crashes involving bicycles is. One possible explanation for this is the low overall number of recorded bicycle accidents and fatal accidents, with the low fatality rate simply coming from an insufficiently large sample. Another explanation is that bicycles are less likely to be on roads with high speed limits, where most fatal accidents occur. To determine the cause more conclusively, a larger data set could be used with an area larger than Vermont and the number of crashes involving bicycles on certain road types and other conditions could be compared to the fatality rate for any crash under said conditions.

Limitations for These Two Graphs

There are two major limitations in these visualizations that result primarily from the data set used. First, some of the categories had very few entries, such as bicycle crashes in windy weather, which only had a single entry. This is mainly due to the data set, particularly for older dates, having many missing values. Many crashes simply needed to be ignored as they were missing information about the type of vehicle they involved, the weather conditions at the time, or both. This could have been ameliorated by grouping all weather conditions into a single bar and not ignoring entries without a weather condition listed, but I decided that the increase in information displayed by showing the weather was worth having to remove a few of the sections that did not have enough entries.

The second limitation was the lack of whether or not multiple types were involved in a crash (i.e, pedestrian and heavy truck rather than just pedestrian). While it's obvious that crashes involving a pedestrian are going to be more fatal than one involving just a heavy truck, with the data set used, information such as how the injury and fatality rates of accidents with pedestrians are affected by whether the accident also involve a bike, motorcycle, or heavy truck is not able to be determined.

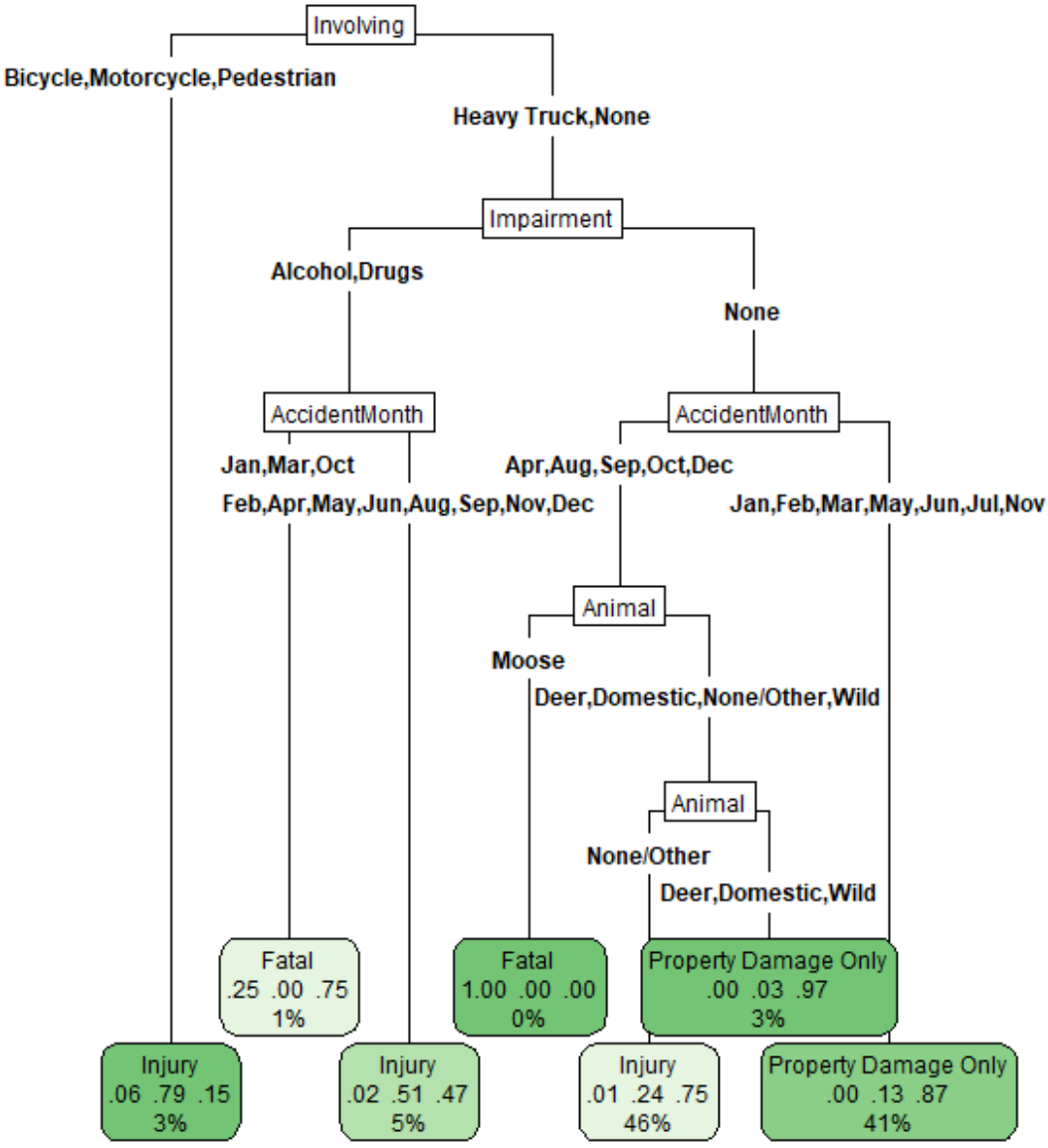
Machine Learning

```
# Select only the columns we will use to predict
crash_data_ml <- crash_data_na %>%

  dplyr::select(ACCIDENTDATE, Animal, Impairment, Involving,
                Weather, InjuryType, SurfaceCondition, DayNight) %>%

  na.omit()


# Format Date
crash_data_ml <- crash_data_ml %>%
```

```r
  mutate(AccidentDate = str_extract(string = ACCIDENTDATE,
                    pattern = "(\\d\\d|\\d)/(\\d\\d|\\d)/(\\d\\d\\d\\d)"))
%>%

  mutate(AccidentDate = mdy(AccidentDate)) %>%

  mutate(AccidentMonth = month(AccidentDate)) %>%

  mutate(AccidentMonth = case_when(
    AccidentMonth == 1  ~ "Jan",
    AccidentMonth == 2  ~ "Feb",
    AccidentMonth == 3  ~ "Mar",
    AccidentMonth == 4  ~ "Apr",
    AccidentMonth == 5  ~ "May",
    AccidentMonth == 6  ~ "Jun",
    AccidentMonth == 7  ~ "Jul",
    AccidentMonth == 8  ~ "Aug",
    AccidentMonth == 9  ~ "Sep",
    AccidentMonth == 10 ~ "Oct",
    AccidentMonth == 11 ~ "Nov",
    AccidentMonth == 12 ~ "Dec",
  )) %>%

  dplyr::select(- c("ACCIDENTDATE", "AccidentDate"))


# Convert all strings to factors
crash_data_ml <- as.data.frame(unclass(crash_data_ml),
                               stringsAsFactors = TRUE)

crash_data_ml <- crash_data_ml %>%

  mutate(AccidentMonth = factor(AccidentMonth,
                                levels = c("Jan", "Feb", "Mar", "Apr",
                                           "May", "Jun", "Jul", "Aug",
                                           "Sep", "Oct", "Nov", "Dec")))

RNGversion("4.0.0")
set.seed(1234)

# Generate the split
split <- sample.split(crash_data_ml$InjuryType,
                      SplitRatio = 0.7)

train_set <- crash_data_ml[split,]

test_set <- crash_data_ml[!split,]

table(train_set$InjuryType)
```

```
##
##                Fatal               Injury Property Damage Only
##                  473                16055                58428

table(test_set$InjuryType)

##
##                Fatal               Injury Property Damage Only
##                  203                 6881                25041

# Create the cost matrix
cost_df <- tribble(
  ~Actual,                ~Predicted,             ~Cost,
  "Property Damage Only", "Property Damage Only", 0,
  "Property Damage Only", "Injury",               1,
  "Property Damage Only", "Fatal",                1,
  "Injury",               "Property Damage Only", 5,
  "Injury",               "Injury",               0,
  "Injury",               "Fatal",                1,
  "Fatal",                "Property Damage Only", 5,
  "Fatal",                "Injury",               3,
  "Fatal",                "Fatal",                0
)

cost_mat <- xtabs(formula = Cost ~ Actual +  Predicted, data = cost_df)

# Create and prune the tree
crash_tree_full <- rpart(formula = InjuryType ~ .,
                         data = train_set[1:1000,],
                         method = "class",
                         parms = list(split = "information",
                                      loss = cost_mat),
                         minsplit = 0,
                         minbucket = 0,
                         cp = -1)

crash_tree_pruned <- prune(crash_tree_full,
                           cp = 0.009)

# Evaluate the tree
train_pred <- predict(object = crash_tree_pruned,
                      newdata = train_set,
                      type="class")

test_pred <- predict(object = crash_tree_pruned,
                      newdata = test_set,
                      type="class")

cm_train <-
  confusionMatrix(data = train_pred,
```

```
                  reference = train_set$InjuryType,
                  positive = "Property Damage Only",
                  dnn = c("Predicted", "Actual"))

cm_test <-
  confusionMatrix(data = test_pred,
                  reference = test_set$InjuryType,
                  positive = "Property Damage Only",
                  dnn = c("Predicted", "Actual"))

# Print accuracy of model
paste("The pruned tree had ", round((cm_train$overall["Accuracy"] * 100), 2),
      "% accuracy with the training data set.", sep = "")

## [1] "The pruned tree had 57.32% accuracy with the training data set."

paste("The pruned tree had ", round((cm_test$overall["Accuracy"] * 100), 2),
      "% accuracy with the testing data set.", sep = "")

## [1] "The pruned tree had 57.57% accuracy with the testing data set."

# Plot tree
rpart.plot(x = crash_tree_pruned,
           type = 5,
           extra = "auto",
           box.palette="Greens")
```



Machine Learning Limitations

While trying to perform machine learning on this data set, I immediately ran into the issue that the majority of all crashes recorded caused property damage only. This led to essentially everything being predicted as property damage and a tiny amount being predicted as injury, with none predicted as fatal. Though this was the most accurate of all of the models (around 87%), it only had one level and didn't show us anything about the data that we didn't already know. I used a cost matrix to weight the cost of incorrectly predicting property damage to be 5 times normal and the cost of incorrectly predicting injury when it was fatal to 3 times normal, as in this new tree, fatal was also never being

predicted. Though this had a lower accuracy of around 57%, it revealed more about the data.

To fix some of these problems, multiple trees could have been created, such as one predicting fatal or injury and one predicting property damage or either fatal or injury. Using both of these trees together could give a more accurate result.

Explanation

Even with the problems, several interesting pieces of information are shown in this tree. For instance, there is very little correlation between month and injury type. Though it was used in the tree, there is no similarity (i.e., winter months vs other seasons) between the months used in the tree. While other parts of our data analysis showed a possible link between the month and car crashes, that was a possible correlation with the **number** of crashes, rather than the **severity**, which is what is shown by the tree.

It was also shown that accidents involving moose are more likely to cause fatalities than accidents involving other types of animals.Impairment was show to greatly increase the chance of both injuries and fatalities and was never predicted to be property damage only.

Having pruned the tree, we can also see the most important variables for determining injury type. These turn out to be the type of vehicles / pedestrians involved, whether or not a driver is impaired, whether an animal is involved, and what specific type of animal it was.

I decided to use a classification tree for this data set as we were trying to predict whether a given crash would be one of three things (property damage, injury, or fatality). In addition, most of our variables were non-numeric, ruling out any kind of kNN classification.

Project Limitations

The data included in this research was only for the state of Vermont. Our findings will unlikely be able to be generalized to other states or areas of the country. To do a better study of factors that influence car crashes, a larger area should be used such as all of New England or even for each state. A study at that magnitude would allow the researcher to make more concrete conclusions that are applicable to more people.

Also, this dataset contained a large amount of missing values which may also factor into some limitations. Without all of the information present, we can lose some valuable data to work with. A follow up to this project could be based on the line graph that shows the number of crashes per day from 2010 to 2020. This time frame of 10 years is very large and it makes the graph harder to interpret. A version that included 10 line graphs, each for 1 year could allow more observations to be made. The patterns and cycles would be easier to see.

Conclusion

As mentioned in the introduction, our main goal was to consider whether certain factors affect the number of accidents and the outcomes of those accidents. We learned many things from our visualizations and analysis that helped us start to answer this question.

First, we learned that most car crashes in Vermont take place on highways and around cities. We also found that the distribution of fatal crashes throughout the state of Vermont seemed random. We learned that the most common type of damage or injury is property damage only. Rear end and single vehicle crash are the two most common collision directions. The line graph shows a cyclical pattern of number of crashes throughout the years. The high points are towards the beginning of the year, maybe February or March which suggests that weather, including snow or ice, could impact the number of crashes. We also can see in our graphs that accidents are more dangerous when pedestrians or bikes are involved.