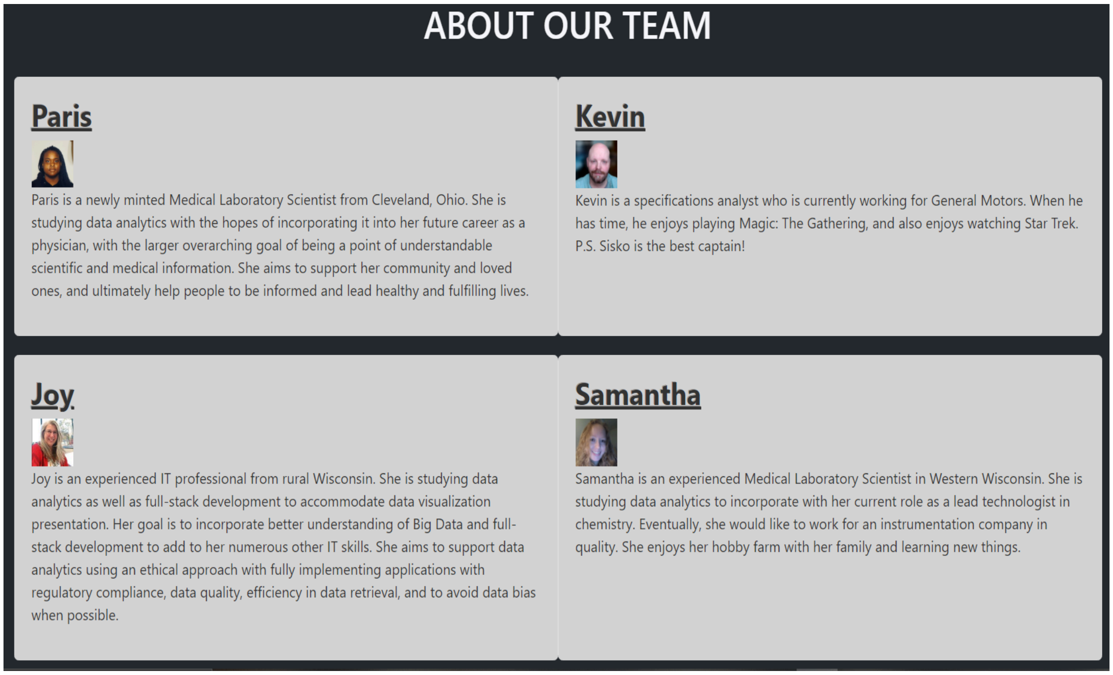
Project 4: Machine Learning Model for Car Crash Data

By: Kevin Carney, Paris Lee, Samantha Schutz, and Joy Weishan
October 3, 2024

Introduction

Driving can be a chore and a source of anxiety for some people. Imagine having the resources to outsource this task. Automated systems are becoming more common in every industry, driving being no exception. It began when Toyota released its first commercial version of parallel park assist in the 2003 Prius. Since then, consumers have been wanting more involved automated driving. The automated systems are still considered to be under development but are currently being utilized by drivers. The question should be “Is it safer?” With these unknowns coming to the forefront the National Highway Traffic Safety Administration (NHTSA) has mandated car manufacturers of these systems to report accidents associated with the use of these systems.

Our group chose this dataset because car crashes are a real risk we take on a daily basis. We find this to be relevant to our lives in addition to a group member working for a car manufacturer. The rest of us work in the medical field and see patients affected by car crashes. In addition to these, we all find artificial intelligence interesting and important for our increased knowledge in data analytics.



ABOUT OUR TEAM

Paris

Paris is a newly minted Medical Laboratory Scientist from Cleveland, Ohio. She is studying data analytics with the hopes of incorporating it into her future career as a physician, with the larger overarching goal of being a point of understandable scientific and medical information. She aims to support her community and loved ones, and ultimately help people to be informed and lead healthy and fulfilling lives.

Kevin

Kevin is a specifications analyst who is currently working for General Motors. When he has time, he enjoys playing Magic: The Gathering, and also enjoys watching Star Trek. P.S. Sisko is the best captain!

Joy

Joy is an experienced IT professional from rural Wisconsin. She is studying data analytics as well as full-stack development to accommodate data visualization presentation. Her goal is to incorporate better understanding of Big Data and full-stack development to add to her numerous other IT skills. She aims to support data analytics using an ethical approach with fully implementing applications with regulatory compliance, data quality, efficiency in data retrieval, and to avoid data bias when possible.

Samantha

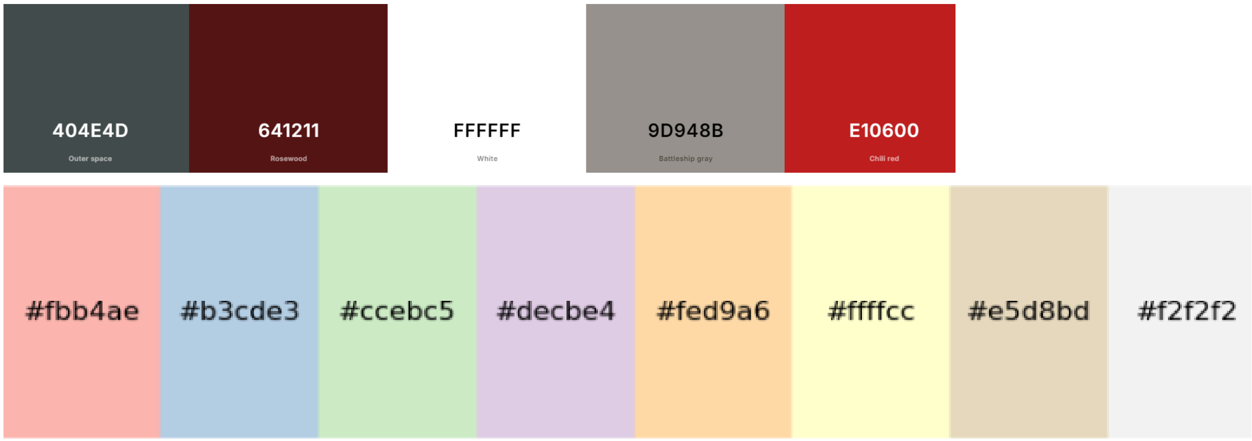
Samantha is an experienced Medical Laboratory Scientist in Western Wisconsin. She is studying data analytics to incorporate with her current role as a lead technologist in chemistry. Eventually, she would like to work for an instrumentation company in quality. She enjoys her hobby farm with her family and learning new things.

The questions we wanted to answer with this dataset included:

1. Will certain weather conditions cause accidents resulting in whether someone gets injured or not? Is this predictable?
2. Will different vehicle conditions impact the result of an accident?
3. When do these accidents happen the most?
4. What are other environmental conditions that impact the outcome of an accident?

Color

We used https://bootswatch.com/slate/ for our website color scheme. The visualizations utilized a wider color scheme shown below.



More colors needed to be added for visualizations because the white could not be used for the visualizations effectively. Four colors were simply not enough for some of the more complex visualizations, so the colors that were used were expanded to accommodate the complexities.

Machine Learning Data Cleaning

As our dataset was pulled from the National Highway Traffic Safety Administration's website, it was not pre-cleaned. Before we could begin making predictions, we needed to understand what our question was that we hoped ML models could predict. Eventually settling on asking "Will certain conditions of an accident result in a higher probability of injury and can this be predicted?".

With our question defined,we wanted to evaluate which of the 137 columns were most likely to have an impact on this prediction. Ultimately we settled on 11 columns of data with an additional two columns that could likely have relevance in future predictions, but did not have any data points within the report. The columns ended up being "Model Year", "Time of Day", "Roadway Type," "Roadway Surface", "Speed Limit," whether passengers were belted, and columns for different weather impacts such as snow or rain. The supplemental columns were for additional weather types of severe wind and fog or smoke.

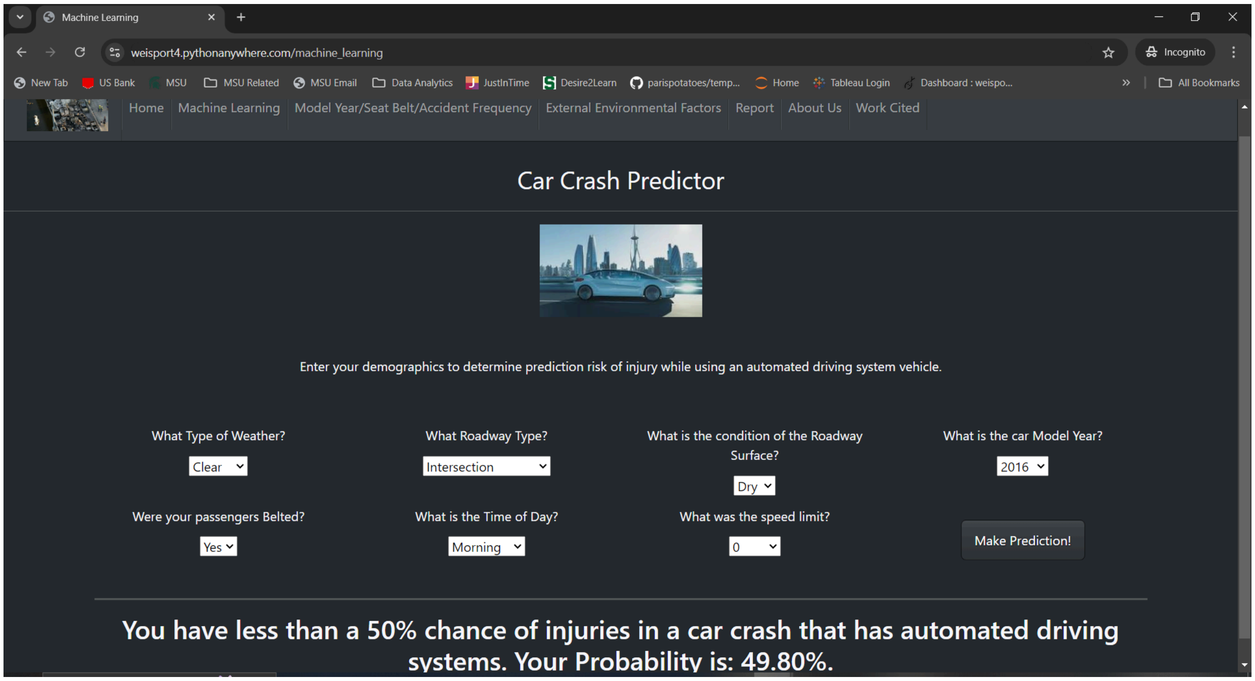
With our data determined, we set about continuing to modify the columns to improve their effectiveness when being loaded into the model. Of particular note was the “Time of Day” column. Whereas the initial dataset had specific times for each of the accidents, we opted to consolidate them down into 3 categories, “Morning”, “Afternoon”, and “Night”. A similar batching approach was applied to both the “Roadway Type” and “Roadway Surface” columns. Lastly we were going to batch the speeds into different “speed categories” but upon rerunning our models found that the performance suffered and as a result, walked back that change.   After the batchings were applied, no other significant data cleaning operations occurred and it was on to testing.

Machine Learning / Interpretations

When it came to testing the model, we ran through many of the common classifiers. Our objective was to find the model that provided the highest weighted average for the F1 Score. The reason for this was because our model is imbalanced, with the number of accidents that didn’t result in injuries greatly outweighing the number of times that injuries occurred.

On average the models were returning with weighted F1 accuracy ratings of about 80%. The best performing model ended up being the XGBoost classifier which had an 83% weighted average F1 Score. However, due to the limitations of the hosting service where these projects were being held, it was felt that the “Random Forest” classifier could also be used. This model also had an 83% weighted F1 accuracy score. However, the false positives / negatives were negligibly worse. The model was then pickled as an h5 file, and used as the basis for the machine learning model in the flask app.

Machine Learning HTML Features



Car Crash Predictor

Enter your demographics to determine prediction risk of injury while using an automated driving system vehicle.

What Type of Weather? Clear

What Roadway Type? Intersection

What is the condition of the Roadway Surface? Dry

What is the car Model Year? 2016

Were your passengers Belted? Yes

What is the Time of Day? Morning

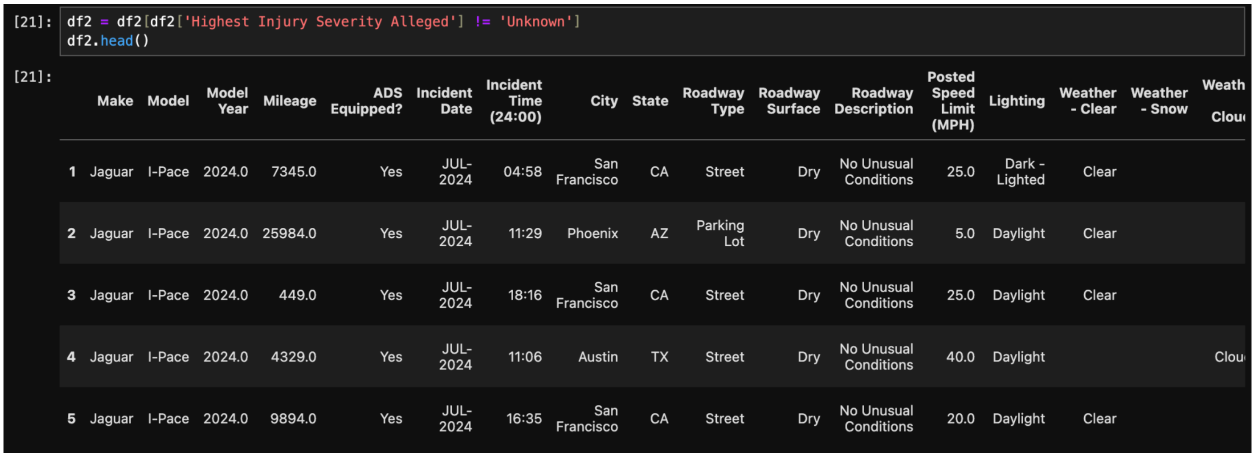
What was the speed limit? 0

You have less than a 50% chance of injuries in a car crash that has automated driving systems. Your Probability is: 49.80%.

Given that our initial question was “Will these different aspects of a crash influence whether injuries occurred or not. It was felt that by enabling a user to effectively describe the scene of a hypothetical accident, the predictive model could then apply its training and return the results accordingly. The model still has difficulties due to the imbalanced dataset, but that functionality is in place. With a bit more time, refining, and/ or data points, the model would likely become much better at making such predictions

Tableau Data Cleaning - Dashboard 1

In doing the second Tableau dashboard, a third Jupyter Notebook was used to clean the dataset. The primary columns of concern were as follows: ‘Highest Injury Severity Alleged', ‘SV Were All Passengers Belted?', ‘Incident Date’, and ‘Incident Time (24:00)’. For the ‘Highest Injury Severity Alleged’ column, unknowns were dropped, as they had no analytic value for the scope of the project. For the ‘SV Were All Passengers Belted?’ column, any row that contained the value ‘No Passengers in Vehicle’ was determined to be an empty car, and thus dropped accordingly. Rows with the value ‘No, see Narrative’ were changed to ‘No’ for the sake of simplicity. The unknowns were also dropped, as they had no analytic value for the scope of the project.

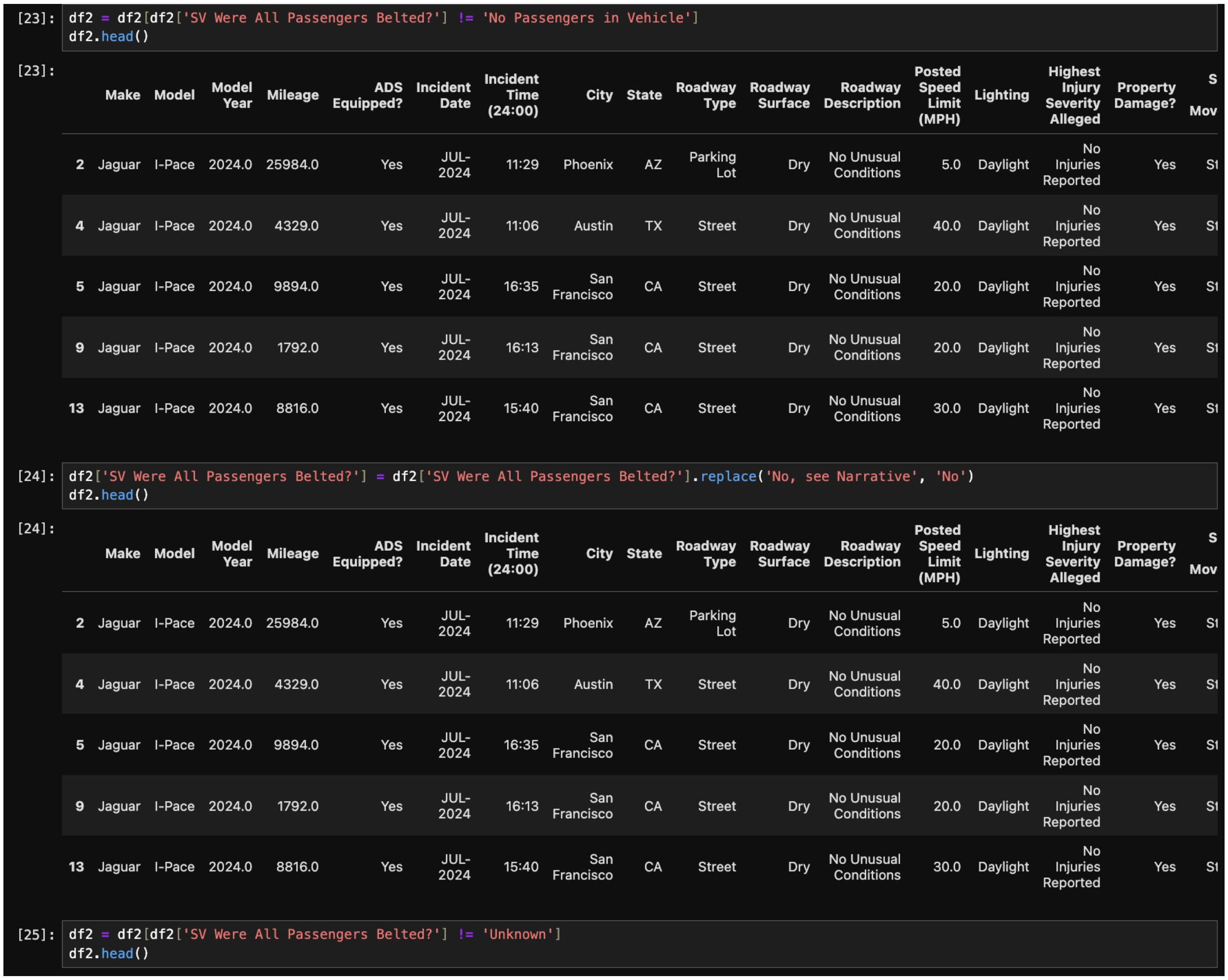


```
[21]: df2 = df2[df2['Highest Injury Severity Alleged'] != 'Unknown']
      df2.head()
```

[21]:

	Make	Model	Model Year	Mileage	ADS Equipped?	Incident Date	Incident Time (24:00)	City	State	Roadway Type	Roadway Surface	Roadway Description	Posted Speed Limit (MPH)	Lighting	Weather - Clear	Weather - Snow	Weath Clou
1	Jaguar	I-Pace	2024.0	7345.0	Yes	JUL-2024	04:58	San Francisco	CA	Street	Dry	No Unusual Conditions	25.0	Dark - Lighted	Clear		
2	Jaguar	I-Pace	2024.0	25984.0	Yes	JUL-2024	11:29	Phoenix	AZ	Parking Lot	Dry	No Unusual Conditions	5.0	Daylight	Clear		
3	Jaguar	I-Pace	2024.0	449.0	Yes	JUL-2024	18:16	San Francisco	CA	Street	Dry	No Unusual Conditions	25.0	Daylight	Clear		
4	Jaguar	I-Pace	2024.0	4329.0	Yes	JUL-2024	11:06	Austin	TX	Street	Dry	No Unusual Conditions	40.0	Daylight			Clou
5	Jaguar	I-Pace	2024.0	9894.0	Yes	JUL-2024	16:35	San Francisco	CA	Street	Dry	No Unusual Conditions	20.0	Daylight	Clear		

Figure 1. Data cleaning to drop rows from the column ‘Highest Injury Severity Alleged’ that contain the value ‘Unknown’.



```
[23]: df2 = df2[df2['SV Were All Passengers Belted?'] != 'No Passengers in Vehicle']
      df2.head()
```

[23]:

	Make	Model	Model Year	Mileage	ADS Equipped?	Incident Date	Incident Time (24:00)	City	State	Roadway Type	Roadway Surface	Roadway Description	Posted Speed Limit (MPH)	Lighting	Highest Injury Severity Alleged	Property Damage?	S Mov
2	Jaguar	I-Pace	2024.0	25984.0	Yes	JUL-2024	11:29	Phoenix	AZ	Parking Lot	Dry	No Unusual Conditions	5.0	Daylight	No Injuries Reported	Yes	St
4	Jaguar	I-Pace	2024.0	4329.0	Yes	JUL-2024	11:06	Austin	TX	Street	Dry	No Unusual Conditions	40.0	Daylight	No Injuries Reported	Yes	St
5	Jaguar	I-Pace	2024.0	9894.0	Yes	JUL-2024	16:35	San Francisco	CA	Street	Dry	No Unusual Conditions	20.0	Daylight	No Injuries Reported	Yes	St
9	Jaguar	I-Pace	2024.0	1792.0	Yes	JUL-2024	16:13	San Francisco	CA	Street	Dry	No Unusual Conditions	20.0	Daylight	No Injuries Reported	Yes	St
13	Jaguar	I-Pace	2024.0	8816.0	Yes	JUL-2024	15:40	San Francisco	CA	Street	Dry	No Unusual Conditions	30.0	Daylight	No Injuries Reported	Yes	St

```
[24]: df2['SV Were All Passengers Belted?'] = df2['SV Were All Passengers Belted?'].replace('No, see Narrative', 'No')
      df2.head()
```

[24]:

	Make	Model	Model Year	Mileage	ADS Equipped?	Incident Date	Incident Time (24:00)	City	State	Roadway Type	Roadway Surface	Roadway Description	Posted Speed Limit (MPH)	Lighting	Highest Injury Severity Alleged	Property Damage?	S Mov
2	Jaguar	I-Pace	2024.0	25984.0	Yes	JUL-2024	11:29	Phoenix	AZ	Parking Lot	Dry	No Unusual Conditions	5.0	Daylight	No Injuries Reported	Yes	St
4	Jaguar	I-Pace	2024.0	4329.0	Yes	JUL-2024	11:06	Austin	TX	Street	Dry	No Unusual Conditions	40.0	Daylight	No Injuries Reported	Yes	St
5	Jaguar	I-Pace	2024.0	9894.0	Yes	JUL-2024	16:35	San Francisco	CA	Street	Dry	No Unusual Conditions	20.0	Daylight	No Injuries Reported	Yes	St
9	Jaguar	I-Pace	2024.0	1792.0	Yes	JUL-2024	16:13	San Francisco	CA	Street	Dry	No Unusual Conditions	20.0	Daylight	No Injuries Reported	Yes	St
13	Jaguar	I-Pace	2024.0	8816.0	Yes	JUL-2024	15:40	San Francisco	CA	Street	Dry	No Unusual Conditions	30.0	Daylight	No Injuries Reported	Yes	St

```
[25]: df2 = df2[df2['SV Were All Passengers Belted?'] != 'Unknown']
      df2.head()
```

Figure 2. Data cleaning to drop empty cars from the ‘SV Were All Passengers Belted?’ column, as well as to replace any ‘No, see Narrative’ values with ‘No’, and to drop the ‘Unknown’ values.

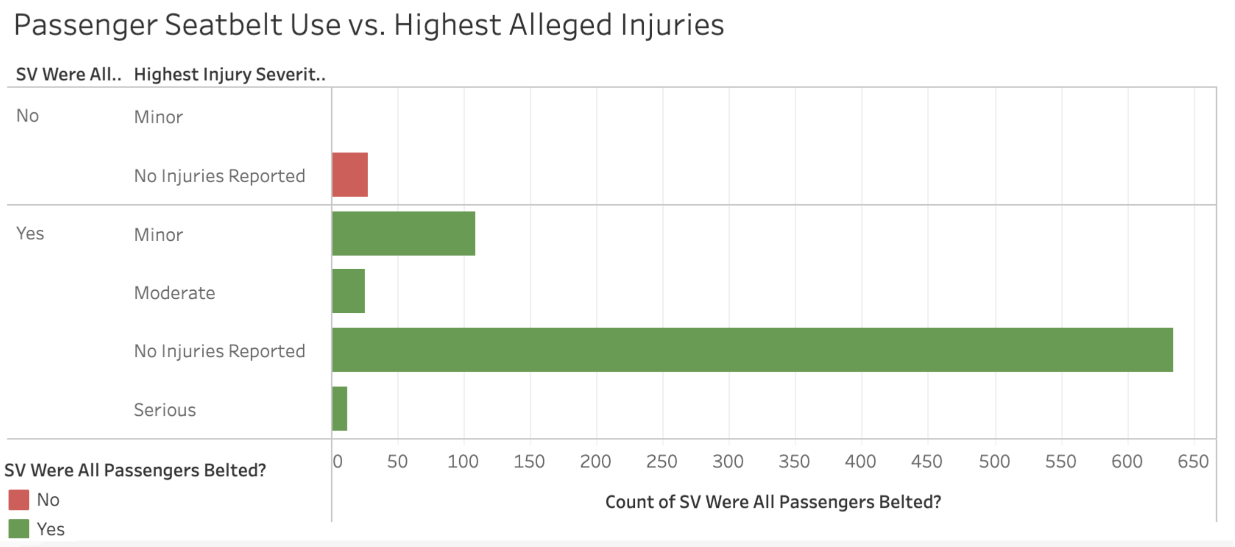


Figure 3. A horizontal bar chart that illustrates the use of seatbelts and the highest alleged injuries.

In this visualization, it appears that there were very few passengers that were not wearing seatbelts, and of those people, an incredibly minor fraction of people not wearing seatbelts sustained even minor injuries. When looking at the data for the passengers that wore seatbelts, the data is a lot more varied, as there is more data for this population. The vast majority of the injuries reported in this group fell under 'No Injuries Reported', a small but notable portion fell under 'Minor' injuries reported, and even smaller for 'Moderate' and 'Serious'. While the insights that can be gleaned from this chart are limited, it does provide an interesting look into the dataset.

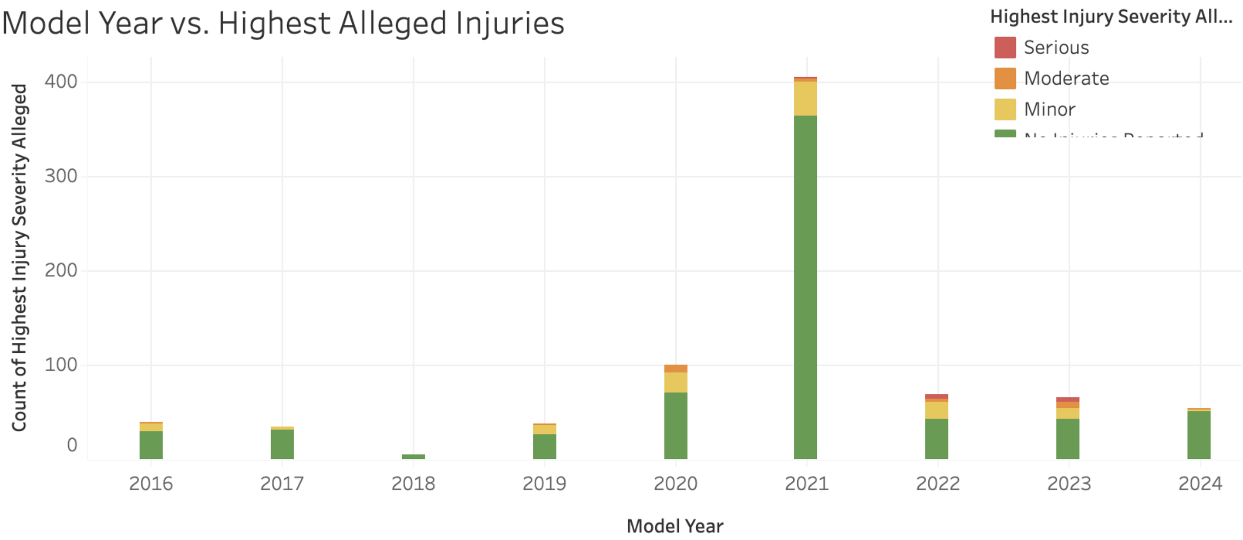


Figure 4. A stacked bar chart that illustrates the model year of the vehicle involved in the accident and counts of the alleged severity of the passengers involved in those accidents.

The goal of this visualization was to test a theory that was proposed: older cars made more predominantly of metal would cause more damage to other cars or passengers (or possibly offer more protection to its passengers) as compared to newer cars that are possibly made predominantly out of plastic. In the chart, it can be seen that the model years of the cars in the dataset span from 2016 to 2024, which challenges the initial theory. That theory was then refuted with the data seen. Most passengers involved in accidents did not report injuries. Injuries reported, from minor to serious, were also an incredibly small fraction of the data, also disproving the initial theory.

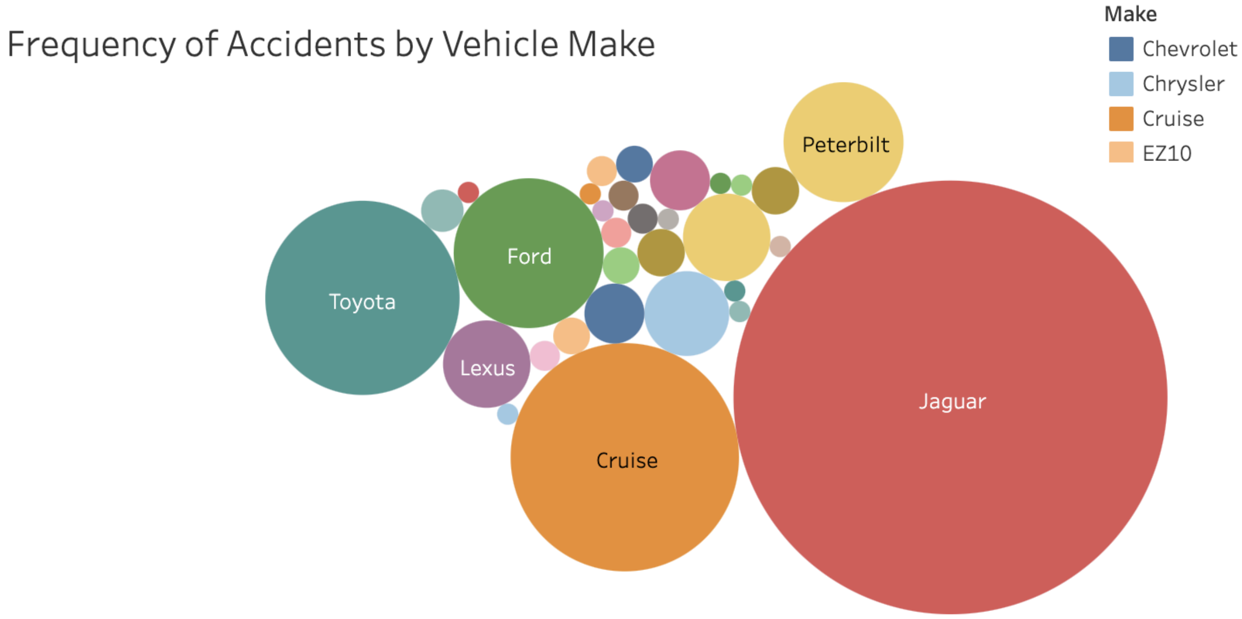
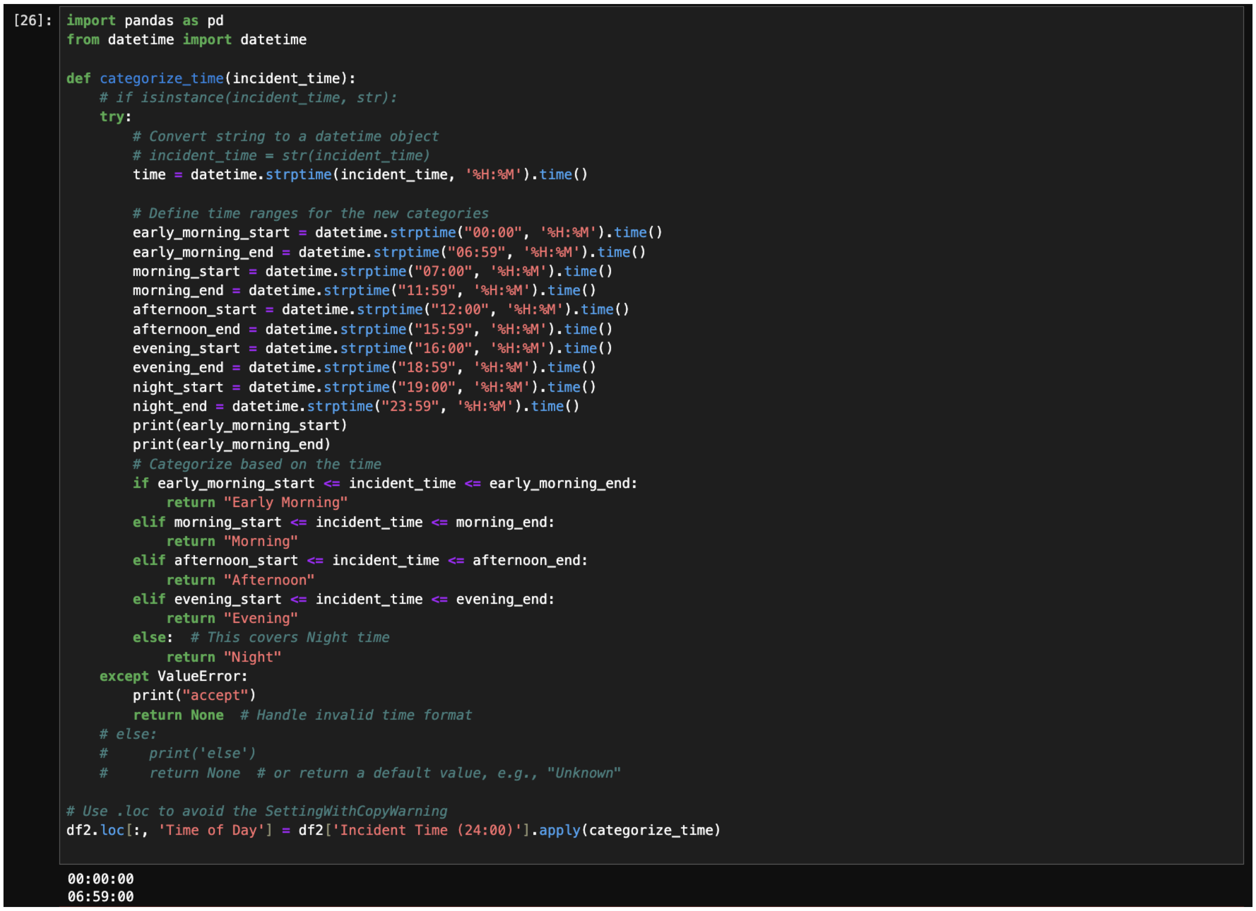


Figure 5. A bubble chart illustrating the frequency of accidents by vehicle make.

With this visualization, it becomes very apparent that the vehicle makes in this dataset are mainly Jaguars, followed by Toyotas and Cruises. This adds to the complexity of the dataset, ultimately influencing any conclusions that can be made from this dataset. Regarding injuries sustained, it might be worth it to factor in safety ratings of the vehicles in question, as that has a very real possibility of having a significant impact on the injuries sustained in a crash. The chart also gives insight into the population of people that were accounted for in this dataset. It is important to note that the vast majority of the data comes from accident data in California, Arizona, and Texas. When examined more closely, overwhelmingly so, most of the data comes from San Francisco.

In attempting to clean the data for the columns ‘Incident Date’ and ‘Incident Time (24:00)’, the column ‘Incident Time (24:00)’ was addressed first. The attempt to categorize the times of the accidents into five columns was made. The columns were to be ‘Early Morning’, ‘Morning’, ‘Afternoon’, ‘Evening’, and ‘Night’. The code for the binning done in the machine learning portion of the project was used and altered accordingly. ChatGPT was used to alter the code for the specified times for the various aforementioned time categories.



```
[26]: import pandas as pd
      from datetime import datetime

      def categorize_time(incident_time):
          # if isinstance(incident_time, str):
          try:
              # Convert string to a datetime object
              # incident_time = str(incident_time)
              time = datetime.strptime(incident_time, '%H:%M').time()

              # Define time ranges for the new categories
              early_morning_start = datetime.strptime("00:00", '%H:%M').time()
              early_morning_end = datetime.strptime("06:59", '%H:%M').time()
              morning_start = datetime.strptime("07:00", '%H:%M').time()
              morning_end = datetime.strptime("11:59", '%H:%M').time()
              afternoon_start = datetime.strptime("12:00", '%H:%M').time()
              afternoon_end = datetime.strptime("15:59", '%H:%M').time()
              evening_start = datetime.strptime("16:00", '%H:%M').time()
              evening_end = datetime.strptime("18:59", '%H:%M').time()
              night_start = datetime.strptime("19:00", '%H:%M').time()
              night_end = datetime.strptime("23:59", '%H:%M').time()
              print(early_morning_start)
              print(early_morning_end)
              # Categorize based on the time
              if early_morning_start <= incident_time <= early_morning_end:
                  return "Early Morning"
              elif morning_start <= incident_time <= morning_end:
                  return "Morning"
              elif afternoon_start <= incident_time <= afternoon_end:
                  return "Afternoon"
              elif evening_start <= incident_time <= evening_end:
                  return "Evening"
              else:  # This covers Night time
                  return "Night"
          except ValueError:
              print("accept")
              return None  # Handle invalid time format
          # else:
          #     print('else')
          #     return None  # or return a default value, e.g., "Unknown"

      # Use .loc to avoid the SettingWithCopyWarning
      df2.loc[:, 'Time of Day'] = df2['Incident Time (24:00)'].apply(categorize_time)

      00:00:00
      06:59:00
```

Figure 6. ChatGPT code to bin the time values in the ‘Incident Time (24:00)’ column.

In this process, multiple errors were encountered. As the troubleshooting took place, it became apparent that the datatype for the values in the column of interest were not in datetime format, but rather just objects. Attempts were made to change the datatype for the values, but it was determined to be unnecessary, and the visualization was created in an alternative manner.

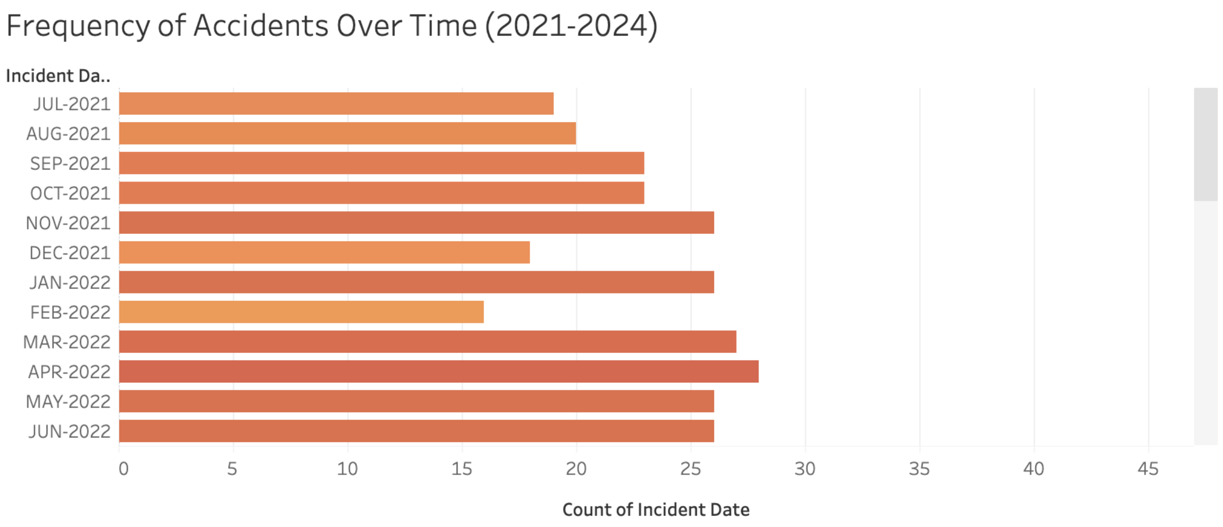


Figure 7. A horizontal bar chart that displays the frequency of accidents over the course of the dataset.

In Tableau, the visualization was created by plotting a count of the incident dates – allowing for the elucidation of the number of accidents that took place every month – against the 'Incident Date' column. This allowed for trends, if present, to be observed over the course of the dataset. A few interesting findings presented themselves, the most notable of which was that it seems that the winter months appear to have less accidents taking place, whereas summer months have more accidents.

Tableau Data Cleaning - Dashboard 2

A different notebook was used to clean the data for the Tableau portion for the "External Environmental Factors" tab to keep more of the columns to make more interesting visualizations. Out of the 137 columns available, all but 26 columns were dropped. There were 1362 rows containing data originally. The remaining columns had the rows with null values dropped, leaving 1290 rows of data. This leaves plenty of data to work with. Of the 26 columns, 6 were individual columns specifying if the weather was a specific condition. For instance, "Weather - Clear" was a column that would be marked Y or left with a space in order to prevent a null value placement. In order to make tableau visualizations out of the weather data, the Y was changed to what weather condition corresponded with the column. The columns were then merged into one column labeled "Weather." Any rows of weather with multiple conditions were changed to one that made sense. For instance, if a weather condition had both rain and cloudy as a condition, just the rain was kept. Columns containing nothing but a blank were considered "clear" for the sake of simplicity. There were many blank spacing in the dataset in the different weather condition columns (I am assuming to prevent null values in this checkmark type reporting). Since spaces are considered an input of sorts, stripping was necessary to take out spacing so the types of weather matched exactly and could be grouped together for visualizations. A CSV was then created to use in Tableau.

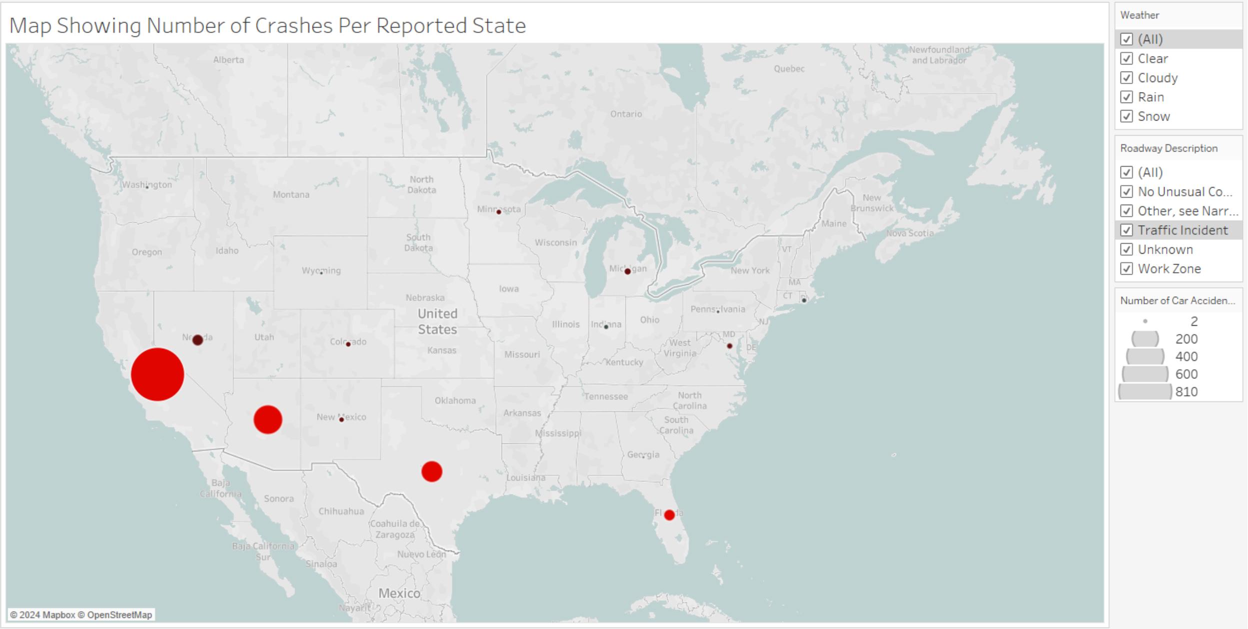


Figure 8: The map shows the number of reported crashes by state. It is a simple visual due to a lot of the location specific information being sanitized for the privacy of those involved in the crashes. Knowing the number of crashes per state can help with prioritizing state specific resources and what needs to be changed to prevent these crashes from occurring. For instance, having only 2 crashes in Michigan would not necessarily warrant any change or resource allocation unless severity was critical.

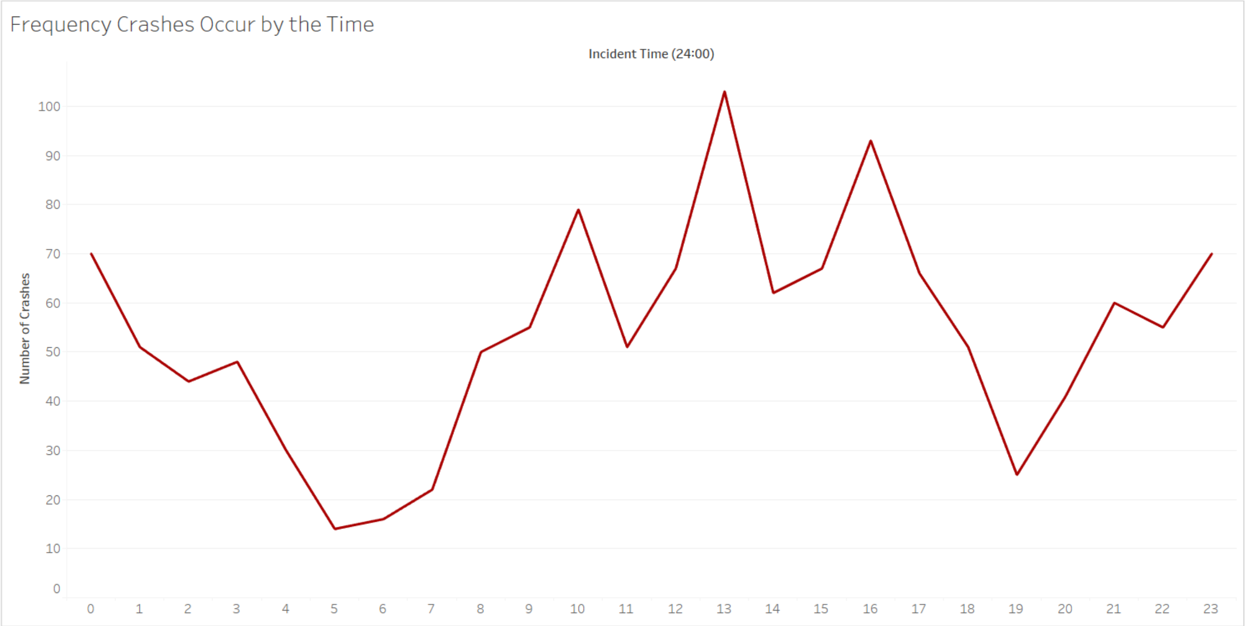


Figure 9: This visual just shows the frequency of crashes by time of day. It makes sense during the middle of the day because generally this is the peak of accidents in any crash dataset and more people are out and about commuting. What is interesting is the influx in the middle of the night. This may be after hours drinking and people utilizing their automation in order to justify

driving if they are not in the best condition to do so. Or are low light conditions not developed well enough for automation?

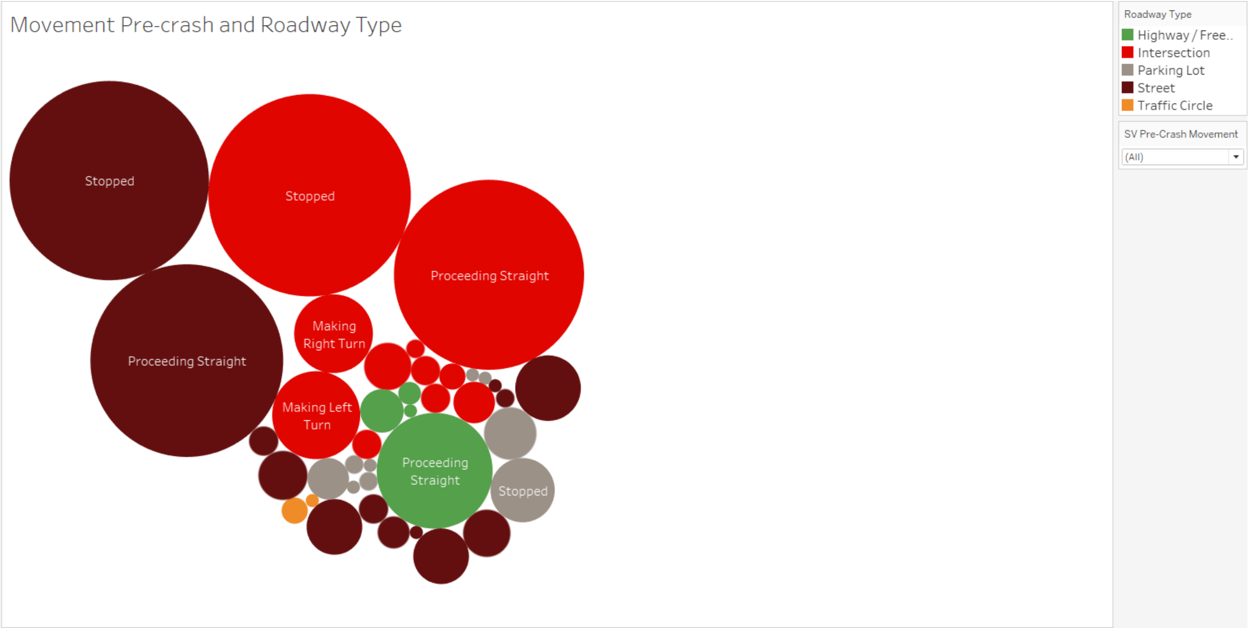


Figure 10: This interactive bubble chart helps to determine what the vehicle was doing when the crash happened. The user can change to filter based on roadway type in addition to specific vehicle movement to figure out if the automated systems may have less functionality in maneuverability.

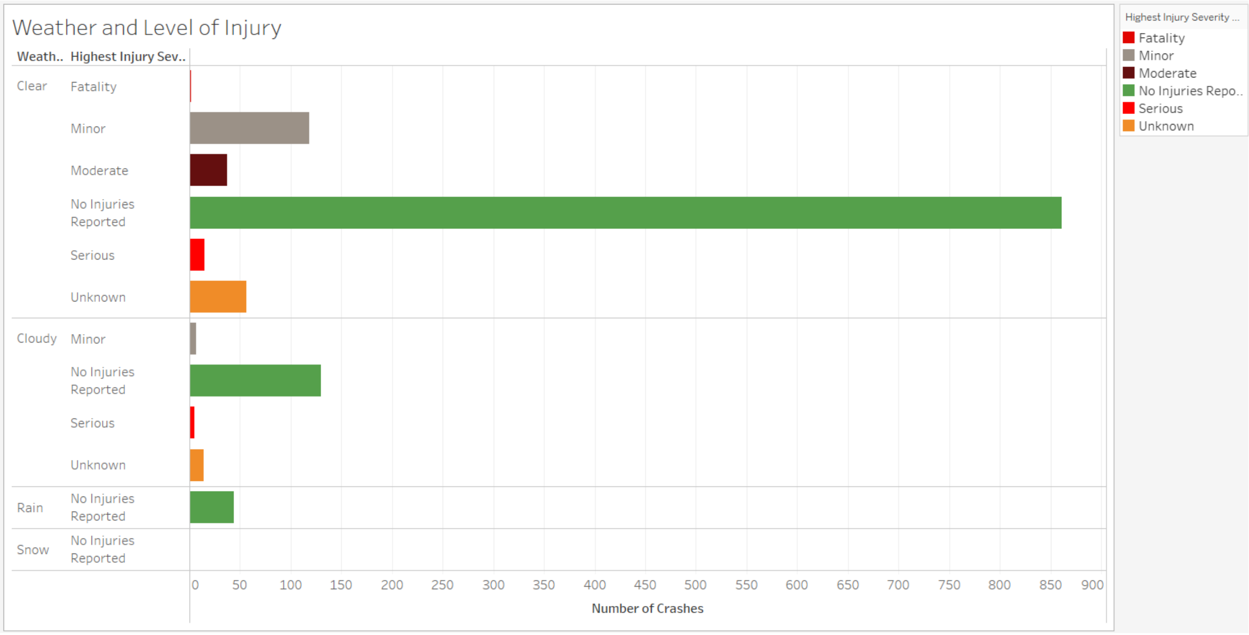


Figure 11: The bar chart shows how many crashes occur and the injury severity based on the weather the crash occurred in. No injuries are the most frequent. Clear conditions are also the

most frequent in this dataset. People using automated systems may be less willing to use the automated system in more severe weather.

The Web App

The Web App uses Flask and was deployed using PCAnywhere.com. The application included information about mandatory reporting for ADS equipped vehicles to the National Highway Traffic and Safety Administration (NHTSA) department, high level history of Automated Data System and the data limitations of the NHTSA mandatory reported data. The HTML also included the Machine Learning Predictive modeling, Tableau Visualizations, access to this report, information about the project team as well as links for any work cited. The Flask App used HTML and the bootstrap slate for the design with style sheets and java script for passing the input data to the machine learning model for crash predictions based on various features.

While deploying the machine learning model in pythonanywhere, the model broke. We found it to be a limitation of the free services compared to the request we made. The model works from the app.py in perfect order. All other tabs do work in the pythonanywhere.

Limitations and biases

The dataset was limited in that many states were not part of the dataset. The mandated reporting is a national requirement so there should be data from all states. This could be due to lack of enforcement of the mandated reporting for these states or a delay in implementation of these requirements by law enforcement.

Crash data varies by manufacturers and the types of automation systems. While data is very detailed and can be reported quickly as these systems are very advanced, the data is not standardized across manufacturers. Remote transmission capabilities vary also by manufacturer. Data reporting is also difficult as these vehicles are privately owned and reporting is based upon owner reporting. Reporting is completed when the manufacturer is notified not at the time of the crash. Reporting is mandatory upon notification. Initial reporting data may be incomplete as reporting at times occurs prior to data verification. Updated verification data is mandatory reported and therefore, does improve data collection. PII data is respected and therefore limits what is reported. The same crash can have multiple reports as there are multiple entities covered under mandatory reporting. Therefore, the number of reports does not equal the number of accidents or incidents.

While cleaning, there were a few assumptions made while simplifying some of the columns. For example, if there was no specific weather condition listed, it was assumed it was "clear." In addition to that, if more than one condition was noted, the one that made the most sense was kept while the other was dropped. Example would be if it was listed as "rainy" and "cloudy", just the "rainy" condition was kept. This helps reduce the number of unique values in these columns in order to help with visualizations in addition to the model.

In terms of the machine learning model, we used two categories to try to balance the data for the model. With binning whether there were injuries or not, there was still an imbalance with no injuries having far more in frequency even with combining all the other severity together. Binning was also necessary in some of the columns fed into the model to get a more predictive outcome. This may have given better numbers in the model, however, the more you have to manipulate the data, the more you lose some of the nuances that could be hidden.

Conclusion/Further work

Vehicles with automated systems are not currently adding accidents in large numbers to the average six million total car crashes occurring in the US every year. Given the information in the dataset it was difficult for many models to predict due to the heavy imbalance in the outcome of the crashes. There are certain manufacturers with more occurrences of crashes. Most people wear seatbelts having all severities of injuries within the category. Interestingly, no seatbelt use had no injuries to minor injuries. August is considered the month with the most car crashes. This is true for one of the August months available within the dataset by about double of any other month but not the other August. The time of day shows the typical rush hour times have an increased crash occurrences. The middle of the night shows an interesting increase in crash occurrences. This could either show the abilities of the automation to function appropriately in lower light conditions. This could also show people using the automation when they are not in the best condition and use it as a way to drive without actually driving.

Arguably there can be a predictive model made from this dataset whether an accident will come out with injuries or not, however not without manipulating the data to create a better predictive analysis. XGBoost was the model with the best overall F1 scores. This is not surprising considering this is a classification model that this model excels in.

Overall, more questions were brought up than answered. There are so many factors that come into a crash that need to be analyzed as a whole. Further work should include if another vehicle was involved and who was at fault. This is relevant to further development of the automation systems to become available in future versions. Either way, automation can only perform specific functions and don’t currently have the ability to interact with the environment in complex ways like a human can. The one benefit is automation can follow road rules well thus having almost no serious outcomes to the people involved.

References

“Park Assist Technology: A History ” by Yamuna Bindu Nov 11, 2021
https://blog.getmyparking.com/2021/11/11/park-assist-technology-a-history/

“Why August is 'the most dangerous driving month' of the year” Published Monday, July 30, 2018. CTV News.

Xpert Learning Assistant:

- Xpert Learning Assistant - Used for debugging certain codes.

chatGPT:

- chatGPT- Used for debugging certain codes.

“NHTSA Standing General Order on Crash Reporting Dataset Source” DOT of US government. June 2022.

“Using Machine Learning Models to Forecast Severity Level of Traffic Crashes by R Studio and ArcGIS” by Bara’ W. Al-Mistarehi, Ahmad H. Alomari, Rana Imam, Mohammad Mashaqba on April 19,2022.