Name: Jonathan Lawrence

Date: 11/9/2019

Assignment: DSC550 – 11.2: Case Study (FINAL) – Traffic Crashes (Chicago)

Traffic Crashes (Chicago)

This study will seek to analyze factors that may contribute to injury during a traffic crash. The analysis will be performed by analyzing data from 10,000 crash records taken from the city of Chicago, IL.

Source: https://data.cityofchicago.org/Transportation/Traffic-Crashes-Crashes/85ca-t3if

Reviewing the data

The data contained many columns that were not relevant to the goal of this study. I only pulled the columns that I felt could contribute to whether or not an injury would occur during a traffic crash. The column names were long and hard to read to so I renamed them to simpler terms. I also performed a check for null values and happened to not find any.

	Speed Limit	Weather	Lighting	Road	Status	Hour	Day
0	30	CLEAR	DARKNESS, LIGHTED ROAD	DRY	NO INJURY / DRIVE AWAY	0	7
1	30	CLEAR	DARKNESS, LIGHTED ROAD	DRY	NO INJURY / DRIVE AWAY	23	6
2	30	CLEAR	DARKNESS, LIGHTED ROAD	DRY	NO INJURY / DRIVE AWAY	23	6
3	30	CLEAR	DARKNESS, LIGHTED ROAD	DRY	INJURY AND / OR TOW DUE TO CRASH	23	6
4	30	CLEAR	DARKNESS, LIGHTED ROAD	DRY	INJURY AND / OR TOW DUE TO CRASH	22	6

```
The dimension of the table is:  (10000, 7)
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 10000 entries, 0 to 9999
Data columns (total 7 columns):
Speed Limit    10000 non-null int64
Weather        10000 non-null object
Lighting       10000 non-null object
Road           10000 non-null object
Status         10000 non-null object
Hour           10000 non-null int64
Day            10000 non-null int64
dtypes: int64(3), object(4)
memory usage: 390.7+ KB
```

Data Wrangling

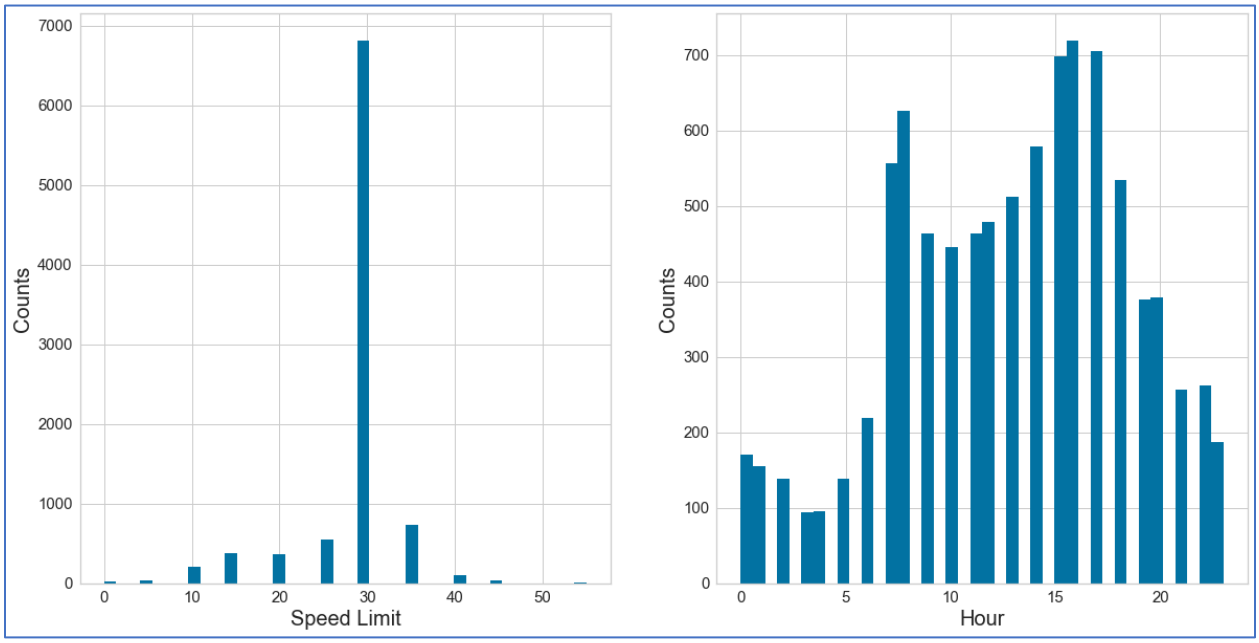
I discovered that the reason why there weren't any null values was because several of the columns had a category for 'UNKNOWN' or 'OTHER' values. I dropped those from the dataset. I also converted the 'Weather', 'Lighting', and 'Road' columns to type 'category'.

```
Unique counts

    Column_Name  Num_Unique
4        Status           2
2     Lighting            5
3          Road           5
6           Day           7
1       Weather           8
0  Speed Limit           15
5          Hour           24
<class 'pandas.core.frame.DataFrame'>
Int64Index: 9252 entries, 0 to 9998
Data columns (total 7 columns):
Speed Limit    9252 non-null int64
Weather        9252 non-null category
Lighting       9252 non-null category
Road           9252 non-null category
Status         9252 non-null object
Hour           9252 non-null int64
Day            9252 non-null int64
dtypes: category(3), int64(3), object(1)
memory usage: 352.8+ KB
None
The dimension of the table is:  (9252, 7)
```

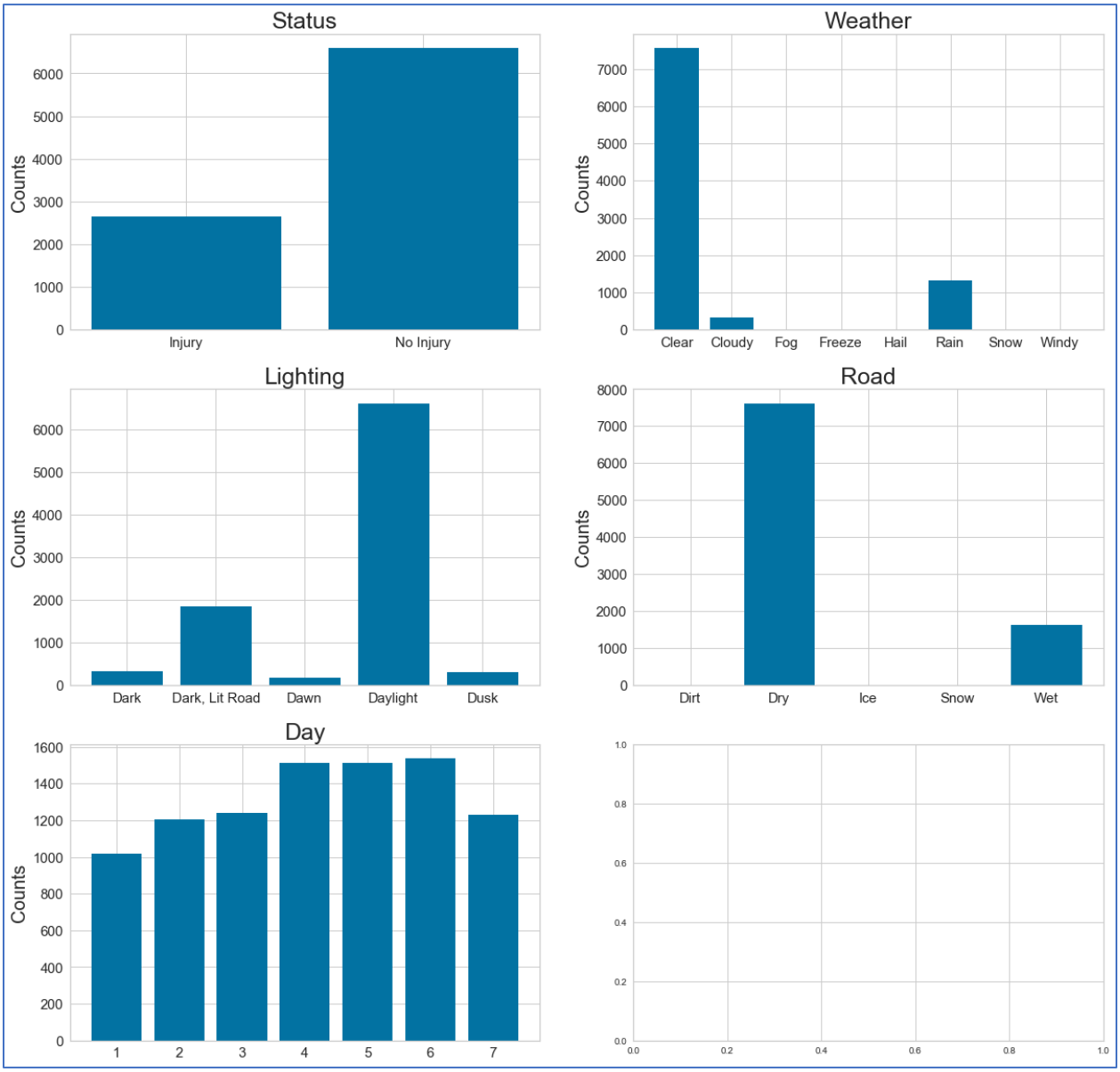
Histograms of Numerical Features

I plotted the speed limit and hours of the day to see if I could notice a trend in the data. The likelihood of a crash occurring grows very slightly with the increase in speed limit, until 30 mph where a vast majority of crashes occur, then it drops off significantly. Given this information, we cannot assume that 30 mph is inherently more prone to crashes, because we do not know if there are an evenly distributed number of roads for each speed limit. It is possible that most roads in Chicago are 30 mph. Crashes occur more frequently during the daytime hours, especially during rush hour traffic between the hours of 7-8 (7-8am) and 15-17 (4-6pm).



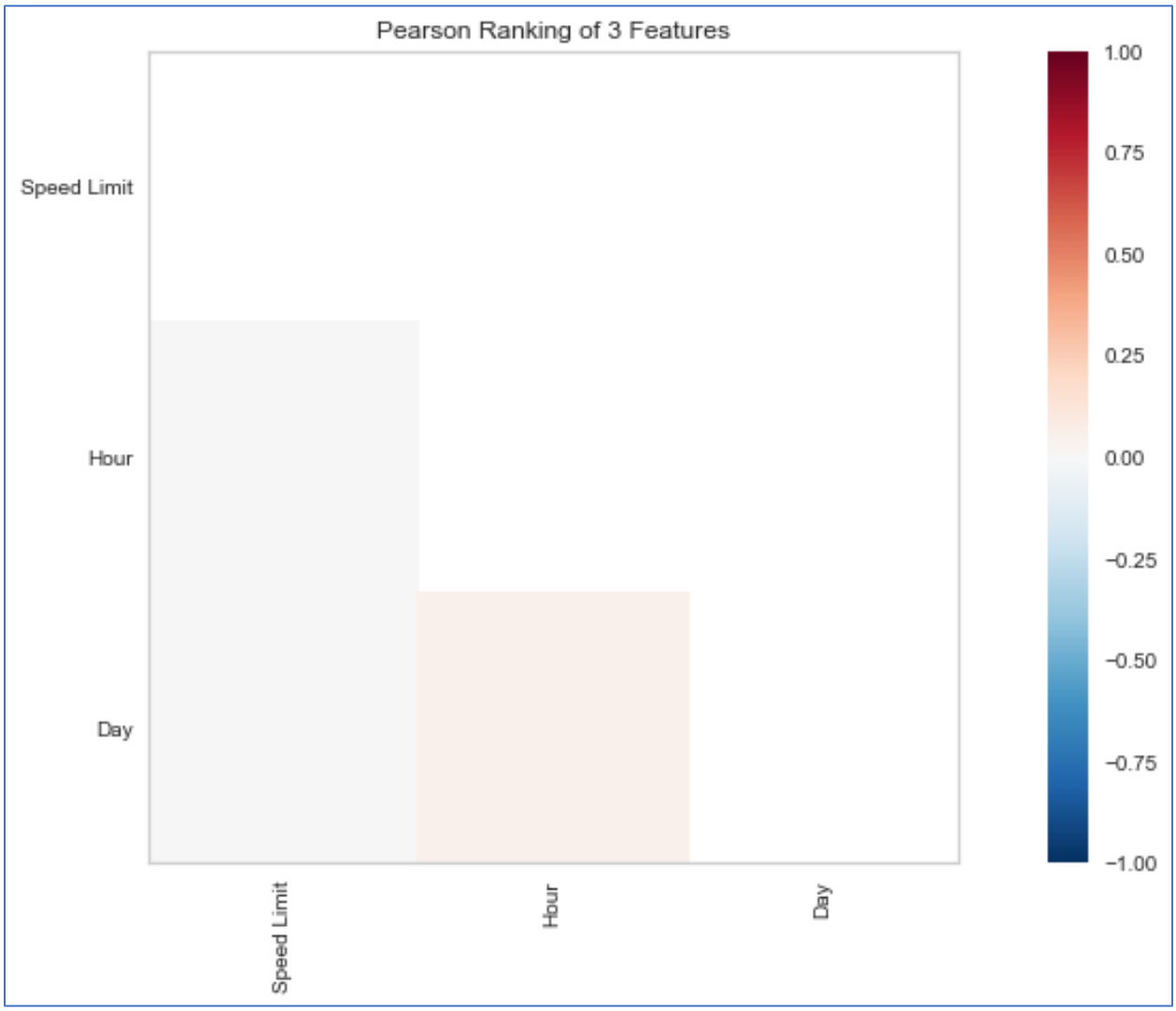
Bar Charts

I plotted some features using bar charts in order to determine how each factor plays out. The insights gained from these charts were a bit surprising. They revealed that most crashes occur without injury, take place during clear weather, during daylight hours, and on dry roads. It also revealed that the chance of being involved in a crash is higher as the week days go on, until the weekend when that chance decreases.



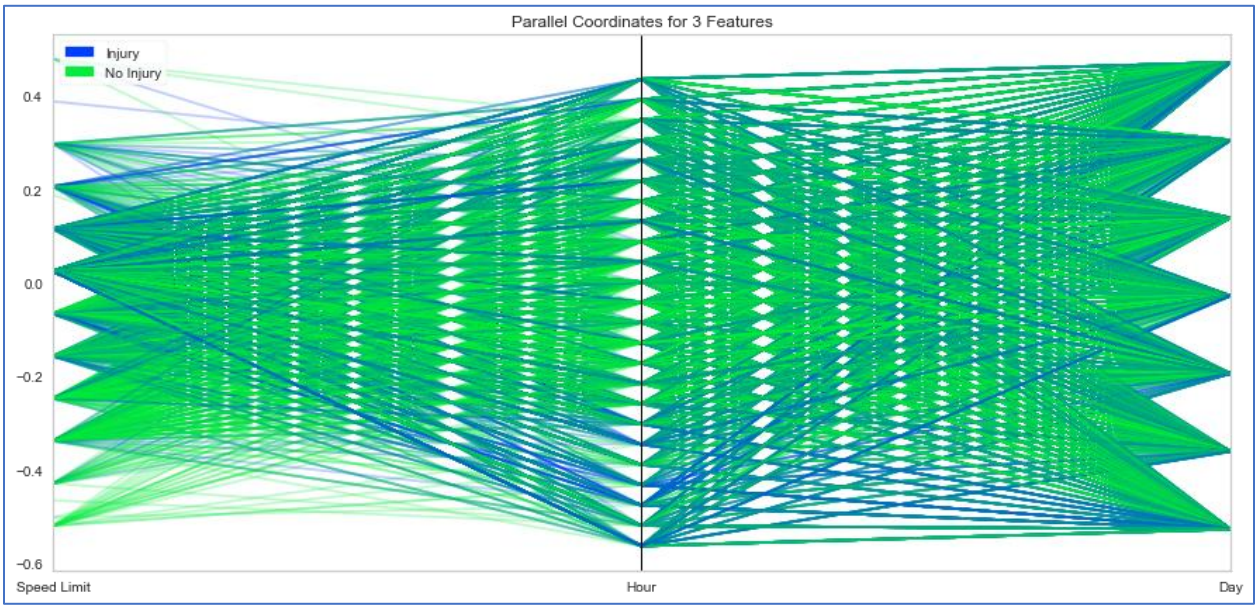
Pearson Ranking

I plotted the Speed Limit, Hour, and Day in a heatmap in order to determine correlation. What I discovered was a very slight correlation between the Hour and Day, as well as between the Speed limit with Hour and Day. The correlation is so small that it barely registers.



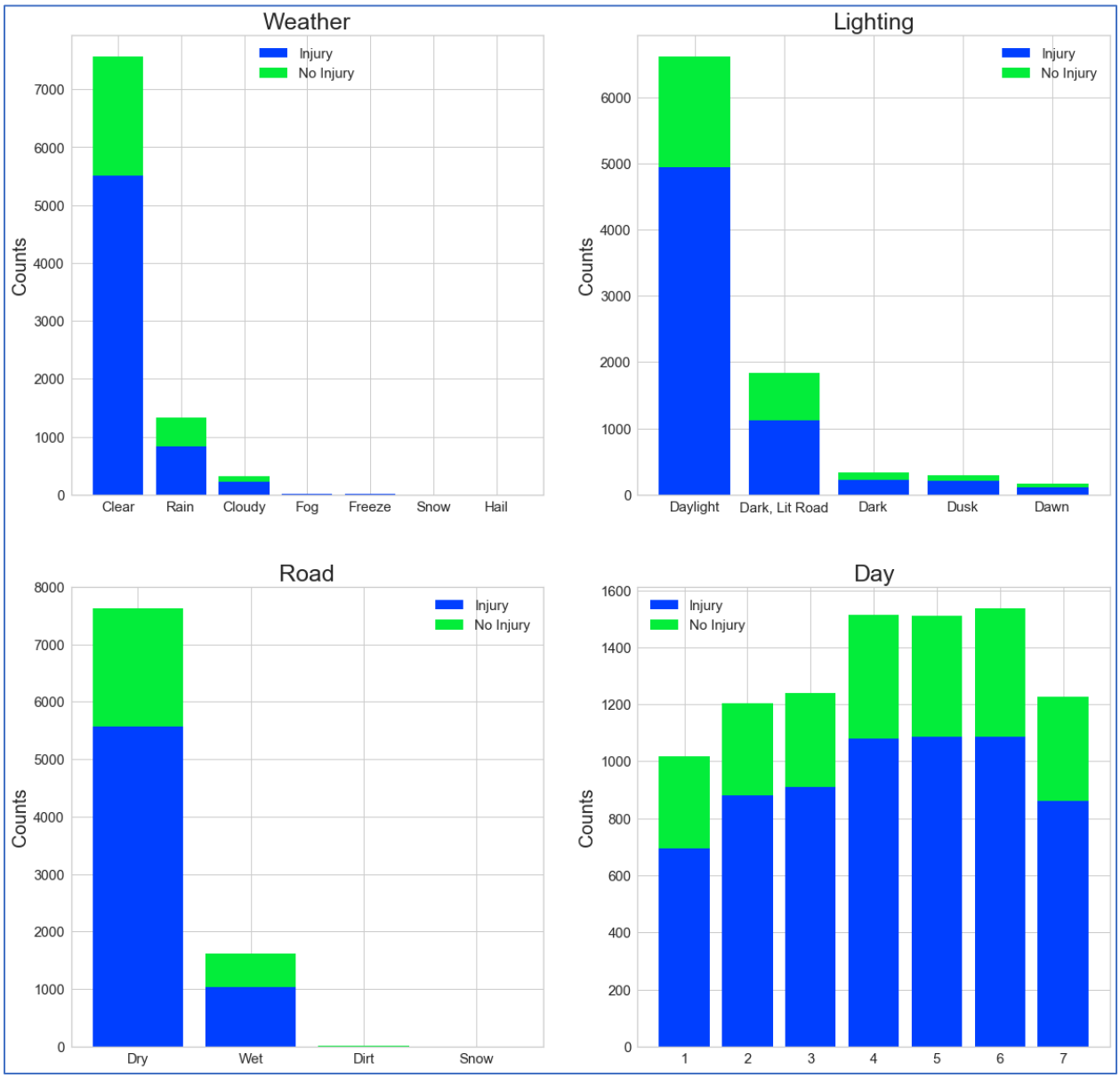
Parallel Coordinates visualization

I created a parallel coordinates visualization so that I could compare different scenarios. I learned that crashed that occurred at slower speeds had a lower chance of injury, while crashes that occurred during rush hour traffic had a higher chance of injury.



Stack Bar Charts

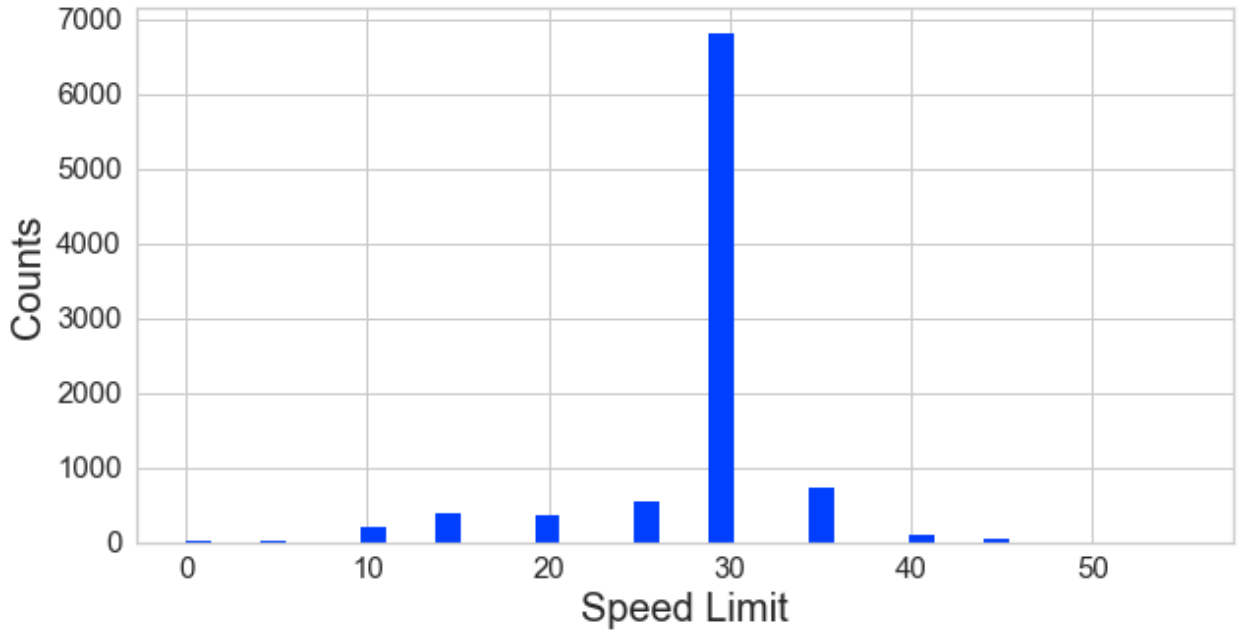
I crated stack bar charts for several key features in order to see if the rate of injury stood out among any of them. From these charts, I learned that most accidents resulted in injury regardless of the weather, lighting, road type, or day of the week. They all seemed to have similar proportions to injury vs no injury.



Feature Reduction

It's at this point that I reviewed the data to see if there was any place that I could further reduce the features. I had already reduced them before starting Part 1 by removing a majority of the columns because they were not useful, like RD_NO, CRASH_DATE, FIRST_CRASH_TYPE, etc. I had also removed all rows with missing values prior to starting Part 1, so there were none in my dataset.

I also considered taking the log transformation of any skewed, ordinal variables. However, the only ordinal variable I am considering is 'Speed Limit' and from the histogram it appears to be normally distributed, and not skewed. So no log transformation is needed here.



One-Hot Encoding

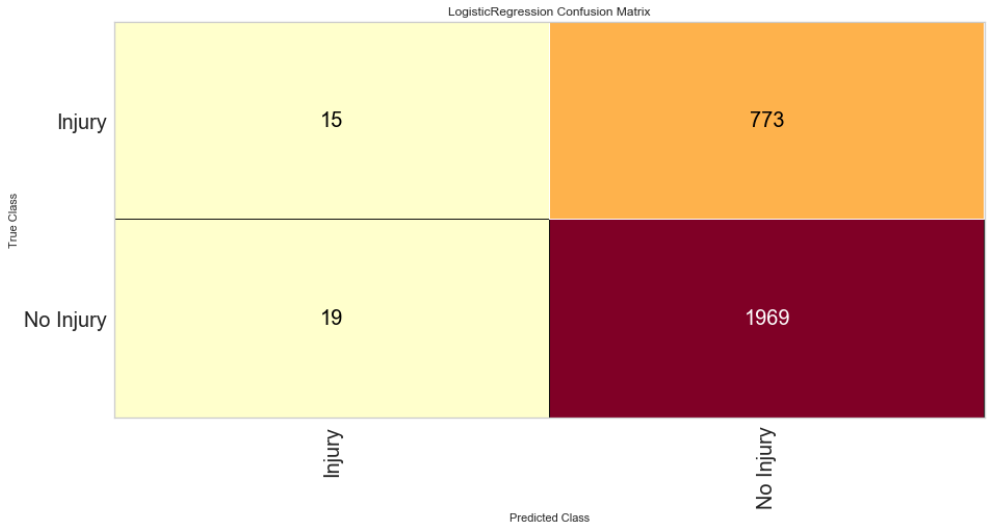
I've used one-hot encoding to convert the categorical variables to 1's and 0's. None of the values are ordinal. I created a new column for each category and assigned a value of 1 or 0. This has made the inputs easier to use for machine learning algorithms. Also, none of the categories are multicollinear, so I didn't have to remove any columns to avoid the dummy variable trap.

	Weather_Clear	Weather_Cloudy	Weather_Fog	Weather_Freeze	Weather_Hail	Weather_Rain	Weather_Snow	Weather_Windy
0	1	0	0	0	0	0	0	0
1	1	0	0	0	0	0	0	0
2	1	0	0	0	0	0	0	0
3	1	0	0	0	0	0	0	0
4	1	0	0	0	0	0	0	0
5	1	0	0	0	0	0	0	0
6	1	0	0	0	0	0	0	0
7	1	0	0	0	0	0	0	0

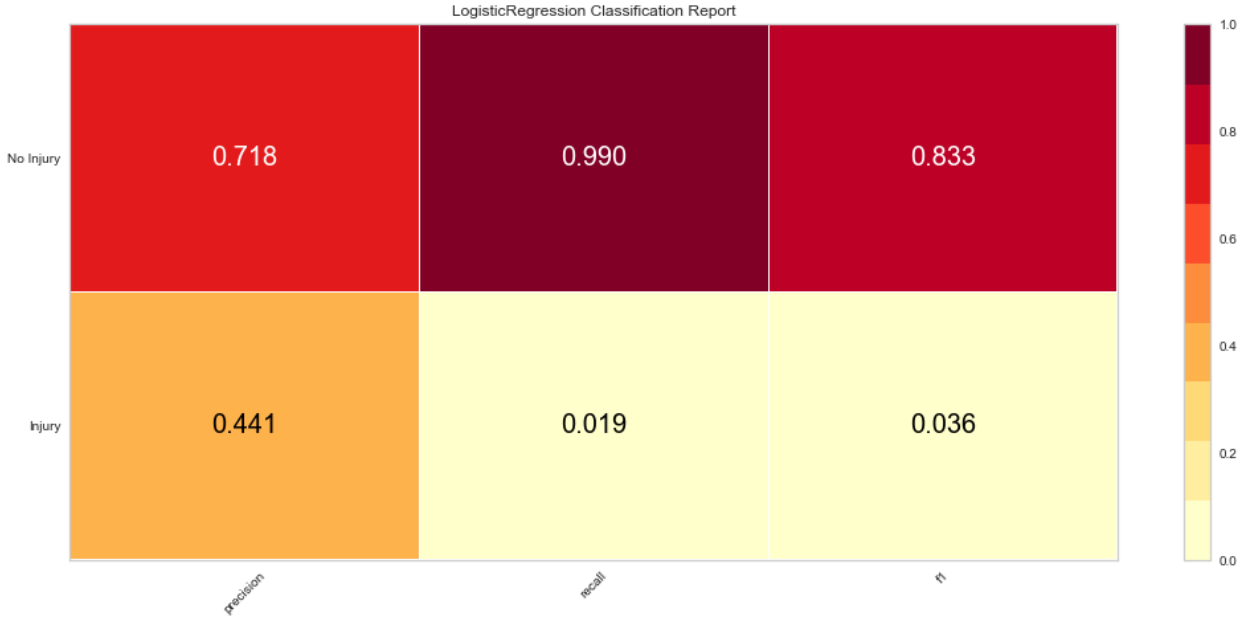
Eval Metrics

In this step I've taken 30% of the data, used it to train my model, and then tested each method on the testing set, so that I could have the machine learning algorithm predict people with and without injury.

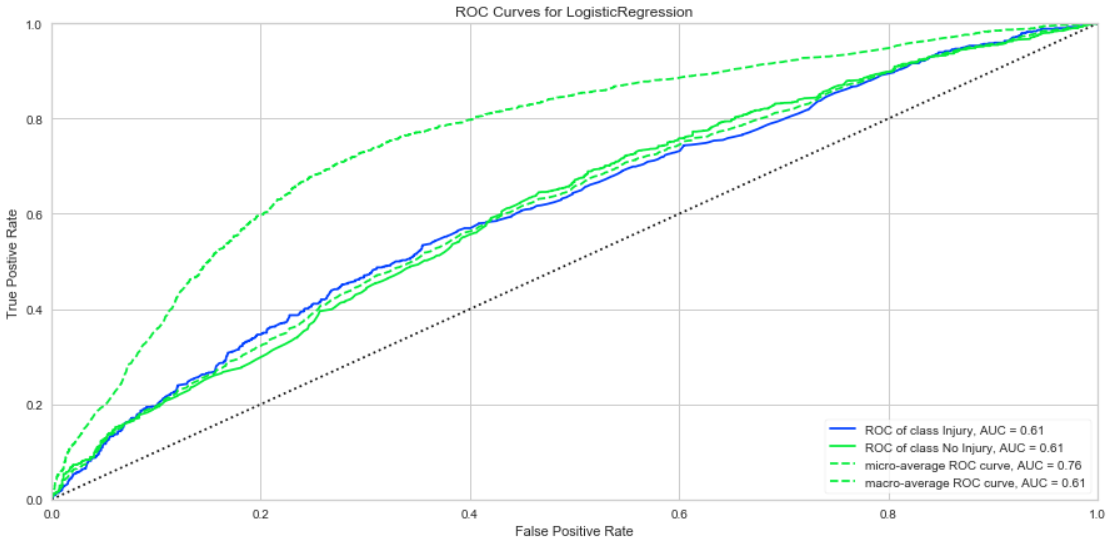
The results show that there are 15 true positives (top-left) for having an injury, that were correctly identified by the algorithm. Alternatively, there were 773 false negatives (top-right) who sustained an injury but the algorithm said they didn't. There were 19 false positives (bottom-left) who did not sustain an injury but the algorithm said they did. Lastly, there were 1,969 true negatives (bottom-right) that did not sustain an injury and the algorithm correctly identified them.



Next, I wanted to be sure that my model was accurate so called the precision, recall, and f1 scores. These values seemed on-par with what I had expected given the previous graphs and calculations.



I plotted the ROC curve (receiver operating characteristic curve) which is a graph showing the performance of a classification model at all classification thresholds. In the ROC curve, we are plotting the difference between the True Positive Rate (percentage of your data points correctly identified as positive) against the False Positive Rate (false identified as positives). From this graph it is apparent that the ROC of Injury and No Injury are mostly the same, and that the values are good but not exceedingly good. The closer the lines are to the top-left of the graph (True Positive with little False Positive) the better.



Conclusion

Overall, the data showed a weak correlation between the features I've chosen and the chance of sustaining an injury during a traffic accident.