Predicting the Severity of a Vehicle Accident

Seth Ofori-Twum

September 15, 2020

**Introduction**

Traffic accidents have always been an ongoing risk and problem since the first commercially manufactured car in the 19th century. As long as cars exist there will always be accidents. These accidents affect many that are involved, whether it is financially, psychologically, or physically, their impact is felt. Being involved in a traffic incident can be traumatic regardless of severity, but wouldn’t it be great if we could predict how bad a vehicle accident would end up being based on factors such as time of day, weather, road conditions, location and boundless other variables that exist in the world? If a driver had a means of knowing if there is a high risk of them getting involved in an automobile accident and how bad the accident would be, then they could easily minimize the risk by choosing not to travel at that time or traveling via a different route. Maybe the previous statement isn’t entirely accurate; Maybe cars can exist without there being accidents through data analysis and implementation of machine learning models, we could make that a thing of the past.

If the City of Seattle municipal government knew what the conditions were that had the most impact before travelling, then it is believed that they would be able to predict the potential risk for an accident and how severe this accident could be during those conditions. In turn, this ability to predict severity of an accident could be used to warn travelers of this risk which would either deter them from travelling during those specific conditions or teach them to travel more cautiously in order to minimize risk. The City could also use this information to have more speed radars in place to encourage safer driving as well as more paramedics on call around the city during these specified conditions.

Accidents cost everyone money and time which for many people isn’t something they can afford to lose. By placing safeguards during these times, the City and its people will save more time, money and possibly their lives just from a few changes in protocol when these conditions occur.

**Data Set: Understanding and Preparation**

Since 2004, Seattle Police Department (SPD), has kept a record of every vehicle collision that has taken place in or around the city. These collisions are recorded at city intersections or in the middle of a block. These incidents could end up being potentially dangerous or even fatal in some cases and are recorded as such with the variable: SEVERITYCODE.

The data comes from Traffic Records Group who collected over 200k cases with information pertaining to the details of the incidents. Each case identified using a coding system where each code represents a description of the incident. After examining and pre-processing the dataset, we can decide whether or not to focus on a specific type of accident or go broad-spectrum and include all codes. Traffic Records Group have recorded many attributes in their data. The most relevant ones that will be worked with include the severity of the accident, which is what our data will try to predict, and the predictor variables: the weather, the road conditions and the light conditions. Table 1.1. gives a brief description of each variable used as shown in the dataset. With this information we can develop a model to warn the City of Seattle about potential severity of accidents occurring when hazards in the environment are present and proceed with the best solution knowing that information.

|  |  |
| --- | --- |
| Attribute (Independent Variable) | Description |
| SEVERITYCODE | A code that corresponds to the severity of the collision: 1- property damage, 2 -bodily injury |
| WEATHER | Describes what the weather was like during the time of the accident:   * Clear * Raining * Overcast * Snowing * Etc. |
| ROADCOND | Describes the condition of the road during the time of accident:   * Dry * Wet * Ice * Snow/Slush * Etc. |
| LIGHTCOND | Describes the lighting/visibility of the road during the time of accident:   * Daylight * Dark- Street Lights On * Dusk * Dawn * Etc. |

The full dataset for the analysis can be found at: <https://s3.us.cloud-object-storage.appdomain.cloud/cf-courses-data/CognitiveClass/DP0701EN/version-2/Data-Collisions.csv> and the metadata that breaks down the variables of the data set comes from here: <https://s3.us.cloud-object-storage.appdomain.cloud/cf-courses-data/CognitiveClass/DP0701EN/version-2/Metadata.pdf>

**Methodology**

Data Set Cleaning

The medium used to perform analysis on this dataset was Python on Jupyter Notebooks. Before diving in and using the data from the dataset, it had to be cleaned up first. All variables were converted into the float datatype, NaN values were removed completely from the set, and rows with variables such as: ‘Unknown’, or ‘Other’ were also removed because we only want to determine severity based on variables that are known during the incidents.

The data was then analyzed to see if there was any correlation to them. Fig.1.1, Fig 1.2, and Fig 1.3 describe the relationship between condition and the number of accidents during each condition.

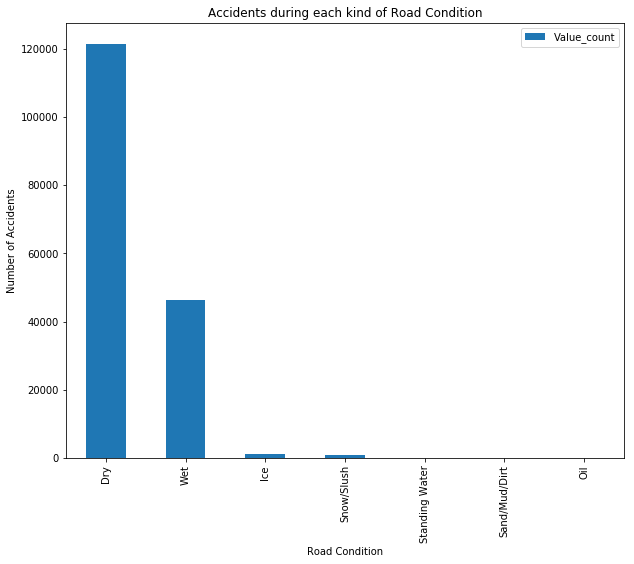


Fig.1.1 Shows the relationship between the number of accidents and the road condition

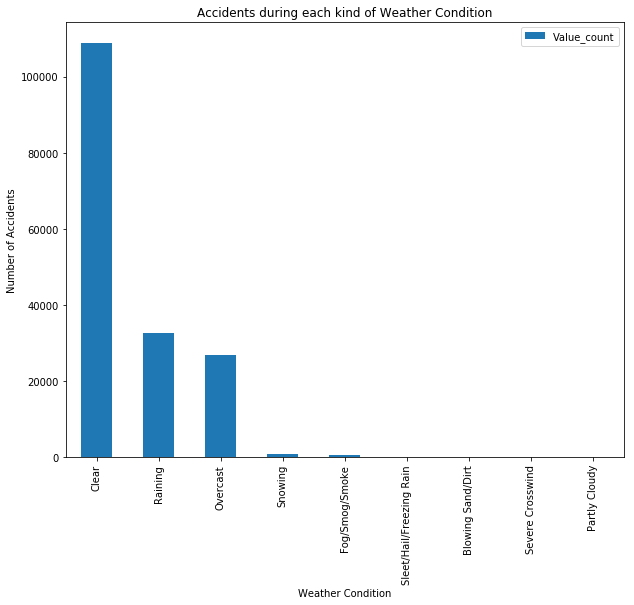


Fig.1.2 Showing the relationship between the number of accidents and the type of weather

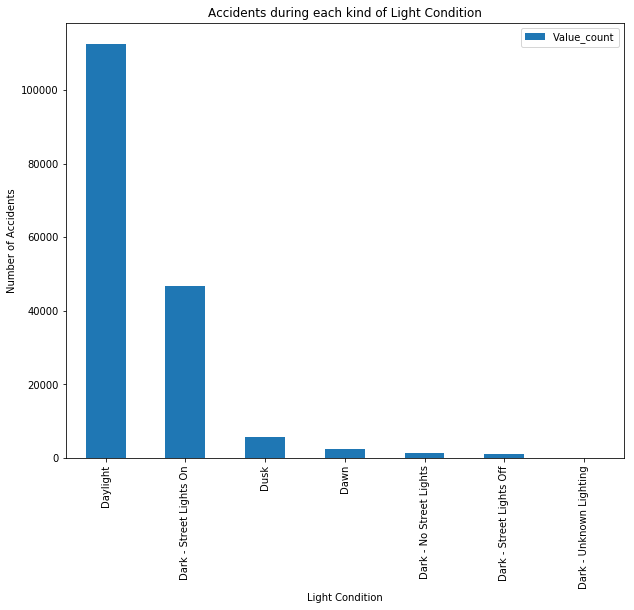


Fig 1.3. shows the relationship between light/ time of day and the number of accidents

Based off the observations, we see that most accidents occur during the daylight, when the weather is clear and the road is dry. The reason for this is possibly because travelers are not expecting to get into an accident during ideal driving conditions and so they are less alert. However more evidence and research would be required to confirm that.

With these factors, we can now predict the severity of the accident using machine learning algorithms.

Model Training

For this data set I trained and tested my model with using three different multivariate algorithms: K-Nearest Neighbors, Decision Trees, and Logistic Regression. All of these methods are practical in their own way so I ran each one to see which gave the most appropriate performance.

Since, it was discovered that the number of accidents was skewed towards property damage, the SEVERITYCODE variable was down-sampled to make each severity count more equal to each other. That way the results would be as unbiased as possible. In order to correct the data to have an even distribution of values, I down-sampled to make both property damage and bodily injury severities equal in weight. This prevents skewness from entering the model’s calculations.

Then, I divided the dataset into a training set where the model learns the data that it will be running for application, and a testing set to actually use to predict previously unknown data. I like to use an 80/20 split which I feel is an appropriate division of the data to train and test the model.

|  |  |
| --- | --- |
| Severity Code | Value Count |
| 1 | 114274 |
| 2 | 55683 |

Table 2.1. Count of severity values before downsample

|  |  |
| --- | --- |
| Severity Code | Value Count |
| 1 | 55683 |
| 2 | 55683 |

Table 2.2. Count of severity values after downsample

K-Nearest Neighbors

A type of classification model that looks to predict a class based on its proximity to similar data points on a graph. It assumes that the points that are close to each or “neighbors” are of a similar class. Using Euclidean distance, we can then calculate the similarity of these points with other points that have other neighbors. The algorithm is programmed to be able to calculate similarity of data points and then return a predicted point that it believes to follow that pattern. This is exactly the type of training that was needed.

In our dataset we want to classify an incident based on its severity and its datapoints are placed on the scale using the variables chosen based on the nearest neighbors of the training set with the same variables.

Decision Trees

Decision trees are also a classification model where the algorithm looks at a sample of binary classifiers and then trains the data to be divided into a node that contains one category using the criteria of the independent variables. The data trickles down into different categories following the decision branches like a tree until the point satisfies all the variables and ends in its final category. This machine learning method is ideal.

In the case of our dataset. The model looks at the severity (binary choice) and decides which category it will most likely fall under depending on the road condition. Then it looks at the weather and decides which severity it belongs to in this case, then again it looks at the light condition where it finally decides which category to be placed in.

Logistic Regression

Logistic Regression once again is a classification model that classifies data points based off values from the input fields. Logistic Regression is an ideal model because it allows for multiple categorical variables to be used to predict probability outcomes from a binary field. In the case of our dataset, it meets all the criteria for using logistic regression because we need to find probability of one outcome occurring or the other, using multiple categorical variables.

After each test was run, an additional test for accuracy was done to determine which learning model produced the most accurate and repeatable results. The criteria for this was Jaccard similarity score and F1-score.

**Results**

Looking at the accuracy of the models through Jaccard index and F1-scores of the test models, it was found that all machine learning algorithms were very similar in score. However, the decision tree seemed to have the highest score Jaccard index albeit not by much as you can see in Table 3.1. but the logistic regression was highest for F1-Score.

|  |  |
| --- | --- |
| Model | Test Accuracy |
| K-Nearest Neighbors | Jaccard index: 51.52%  F1-Score: 47.91% |
| Decision Tree | Jaccard index: 51.83%  F1-Score: 49.64% |
| Logistic Regression | Jaccard index: 51.46%  F1-Score: 50.08% |

**Discussion**

After looking at the results of the tests, I believe the best choice for machine learning model would be logistic regression as the F1-score is more relevant for this data set because we are using binary dependant variables and the logistic regression had the highest F1-score. Using this model, The City of Seattle can decide what preventative actions they can take when the input the current conditions of the road of a specific time to know the severity of the outcome.

I will say that these results do have some issues with them. The first and most obvious thing that is noticed is that the accuracy of these machine learning models is not the highest. All models hover at around 50% accuracy. There could be many reasons for this: the data was not cleaned enough, I could have included more variables in the manipulation of the dataset, the down-sampling of the data may have an effect on the efficacy of the models, or even my train\_test\_split ratio may not have been ideal.

**Conclusion**

In this report, we looked at a problem and identified what it was and how to approach it, who it could affect and who would benefit from knowing this information. By looking at a dataset and having a working knowledge of the Python language, we are able to open doors and explore new possibilities for prediction and simulation of events that are all around us in the real world. In this case we can prepare travelers for the dangers of travelling by car and alert them when it is believed that the severity of an accident takes a serious turn. The data used provides many insights and could be explored even further to develop even more advanced and accurate algorithms in future studies.