Car accident severity on Seattle.

Final report.

Michael Rodriguez Gamboa.
September, 2020.

A. Introduction.

Can we predict the outcome of a transit accident before it happen? How severe would be? It's a job for a prophet? Well, no. With enough data, data analysis and Machine learning it's possible to know the result before the accident. If you can predict the severity you will help to the transit police, the pertinent authorities and the drivers to prevent accidents when dangerous conditions arise.

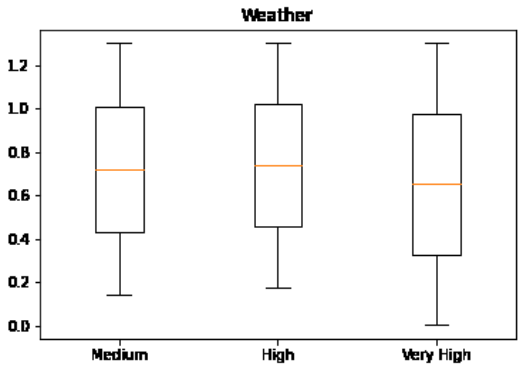
For this work I used a dataset from the weekly transit collision reports of the city of Seattle, USA. These Data contain all the relevant information on transit accidents from 2004 to 2020. I used it to build a predictive model with machine learning.

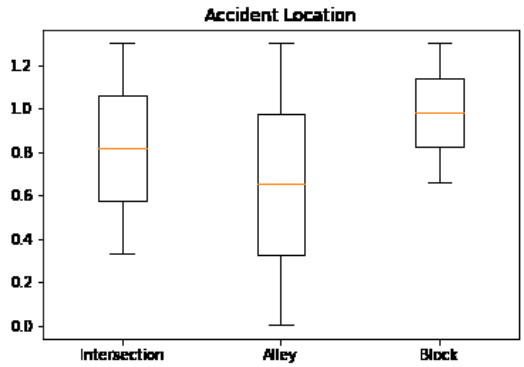
B. Data.

The data used in the analysis is a CVS file with 194,674 reported accidents from 2004 to 2020 in the city of Seattle, USA. The file contains different important details from the incident like: localization, number of cars involved, number of people, condition of the road, light, weather, accident severity and others. The details vary from numerical values, descriptions and categorical classifications. All the columns in the file sum 39. It means that the dimensions in the initial dataset are: 194,674/39.

At first hand, the dataset had missing values in different column, and for that reason I start the analysis dropping the columns with more that 65% of missing data. Others columns with less empty values where filled with the most frequent value, to use it in the data frame.

Then I converted the categorical variables into dummies values: bad weather, road condition, darkness level, and junction type. For the analysis I grouped these features as classification variables from none to very high. Example: badWeather_none, badWeather_low, badWeather_medium, badWeather_high, and badWeather_veryHigh. All this dummy variables can have 0 as false, and 1 as true. In the data set the categorical variables are dominant over the serial types, which will result in a good fit for a Decision Tree algorithm.





The third step in the work was the statistical descriptive analysis, to find the relations between the variables. At this point I decided to use the "severity code" as the target or dependant variable, and 19 features or independent variables as the predictors for the model.

The most important features related to the severity code were:

Features	Person Coefficient & P-Value		Description
Persons Count	Coef= 0.130	p= 0.0	Total of people in the accident.
Pedestrian Count	Coef= 0.246	p= 0.0	Total of pedestrians in the accident.
Bicycles Count	Coef= 0.214	p= 0.0	Total of cyclists.
Vehicles Count	Coef= -0.054	p= 8.177e-129	Total of cars.
Location-Alley	Coef= -0.025	p= 5.210e-30	Where the accident occurs.
Location-Block	Coef= -0.195	p= 0.0	Where the accident occurs.
Location-Intersection	Coef= 0.199	p= 0.0	Where the accident occurs.
Junction-Intersection	Coef= -0.200	p= 0.0	Type of junctions near the accident.
Junction- Mid Block	Coef= 0.200	p= 0.0	Type of junctions near the accident.
Bad Weather- High	Coef= 0.038	p= 1.779e-63	Bad weather during the accident.
Bad Weather- Medium	Coef= 0.014	p= 7.730e-11	Bad weather during the accident.
Bad Weather-Very high	Coef= -0.016	p= 3.828e-13	Bad weather during the accident.
Road condition-Extreme	Coef= -0.011	p= 2.907e-07	Bad Road condition.
Road Condition- High	Coef= -0.020	p= 2.176e-19	Bad Road condition.
Road Condition-Medium	Coef= 0.040	p= 7.743e-73	Bad Road condition.
Darkness- High	Coef= -0.015	p= 1.692e-12	Darkness level.
Darkness- Low	Coef= 0.014	p= 4.449e-10	Darkness level.
At intersection?	Coef= 0.200	p= 0.0	The accident where at intersection?
Under drugs influence?	Coef= 0.044	p= 1.90e-85	The driver was on drugs?

*19 features selected as predictors.

C. Methodology.

After transform in a Numpy array and normalized the data, I split the train and test data from the dataset, using 80% for training. The methodology used for the predict model were a K-Nearest Neighbors and a Decision Tree algorithms. After several tests were found a "k= 4" as the best fit with the KNN algorithm and an "8 depth" level for the D-Tree.

D. Results.

In the model evaluation the test data set was used to determinate the accuracy of the model against new data. The results were good, and the D-Tree model with a level depth equal to 8 obtained 75% of accuracy. There was no sign of under fitting or over fitting with the train data set.

Algorithm	Accuracy	F1-score	Jaccard
KNN	0.7440349300115577	0.7440349300115578	0.7440349300115577
D-Tree	0.7532554257095159	0.7532554257095159	0.7532554257095159

E. Discussion.

In this research, can be obtained relevant information about the elements related to the severity of the car accident. The best predictor found in the data analysis was the location of the car at the moment of the accident. The intersections on the road are the most dangerous place for a accident. The model was able to predict the outcome based mostly in this feature, and others like: Weather, road condition or darkness levels had small impact on the model, but are also useful to predict the severity.

The decision tree had the best results in the prediction, as were expected, with the features been dominant categorical type, and not serial type.

F. Conclusions.

Shall be helpful to the transit police, and the authorities of Seattle use this algorithm to predict the accident severity. The data, after the analysis and the machine learning algorithms, reveled how dangerous are the intersections on the road, and how others variables are not useful to predict the outcome, besides to softly adjust the result. This research would help drivers to prevent accidents, and can guide future researches to discover new information about the severity of the car accidents.

Annexes/Links:

Jupyter notebook.

Presentation.