Predicting the Severity of Car accidents

By

Vipul Gupta

1. Introduction
   1. Background

The Open Data Program makes the data generated by the City of Seattle has been openly available to the public for the purpose of increasing the quality of life for the residents, increasing transparency, accountability and comparability, promoting economic development and research, and improving internal performance management.

The Traffic Records Group, Traffic Management Division, Seattle Department of Transportation, provides data for all collisions and crashes that have occurred in the state from 2004 to the present day. The data is updated weekly and can be found at the [Seattle Open GeoData Portal](https://data-seattlecitygis.opendata.arcgis.com/datasets/5b5c745e0f1f48e7a53acec63a0022ab_0?geometry=-122.326%2C47.592%2C-122.318%2C47.594).

The objective is to exploit this data to extract vital features that would enable us to end up with a good model that would enable the prediction of the severity of future accidents that take place in the state. This would further enable the Department of Transportation to prioritise their SOPs and channel their energy to ensure that fewer fatalities result in automobile collisions.

1. Data

The dataset is available as comma-separated values (CSV) files, KML files, and ESRI shapefiles that can be downloaded from the Seattle Open GeoData Portal. The data is also available from RESTful API services in formats such as GeoJSON.

### Downloading and Loading the Data

We download the dataset to our project directory and take a look at the data types and the dimensionality of the data. We can see that the dataset contains 194673 records and 37 fields.

The metadata of the dataset can be found from the website of the [Seattle Department of Transportation](https://www.seattle.gov/Documents/Departments/SDOT/GIS/Collisions_OD.pdf). On reading the dataset summary, we can determine the description of each of the fields and their possible values.

SEVERITYCODE int64

X float64

Y float64

OBJECTID int64

INCKEY int64

COLDETKEY int64

REPORTNO object

STATUS object

ADDRTYPE object

INTKEY float64

LOCATION object

EXCEPTRSNCODE object

EXCEPTRSNDESC object

SEVERITYCODE.1 int64

SEVERITYDESC object

COLLISIONTYPE object

PERSONCOUNT int64

PEDCOUNT int64

PEDCYLCOUNT int64

VEHCOUNT int64

INCDATE object

INCDTTM object

JUNCTIONTYPE object

SDOT\_COLCODE int64

SDOT\_COLDESC object

INATTENTIONIND object

UNDERINFL object

WEATHER object

ROADCOND object

LIGHTCOND object

PEDROWNOTGRNT object

SDOTCOLNUM float64

SPEEDING object

ST\_COLCODE object

ST\_COLDESC object

SEGLANEKEY int64

CROSSWALKKEY int64

HITPARKEDCAR object

The data contains several categorical fields and corresponding descriptions which could help us in further analysis. We make an attempt at understanding the data in terms of the fields that we shall take into account for later stages of model building.

The X and Y fields denote the longitude and latitude of the collisions.

### Balancing the dataset

Our target variable **SEVERITYCODE** is only 42% balanced. In fact, SEVERITYCODE in class 1 is nearly three times the size of class 2.

We can fix this by down-sampling the majority class.

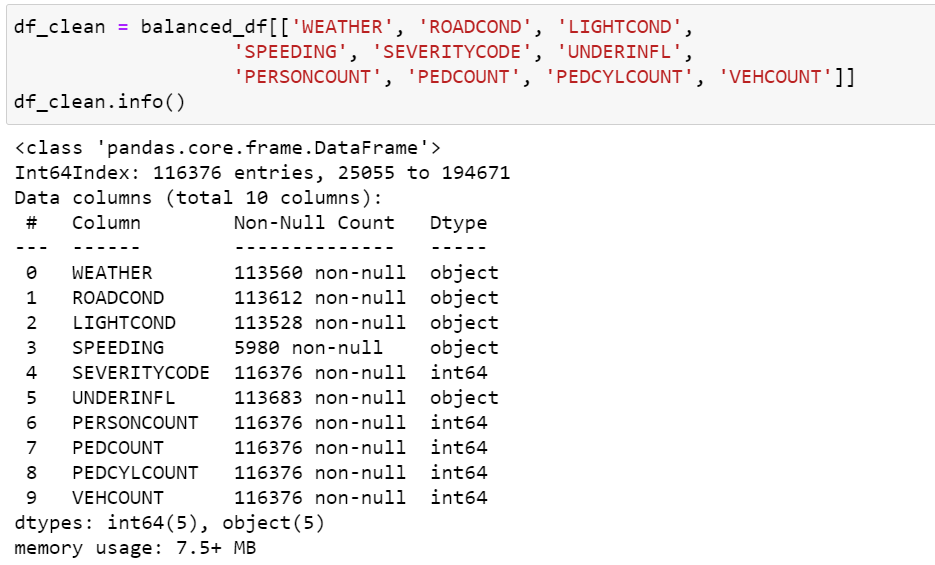
[Alt Text](https://res.cloudinary.com/practicaldev/image/fetch/s--ab3RblIA--/c_limit%2Cf_auto%2Cfl_progressive%2Cq_auto%2Cw_880/https:/dev-to-uploads.s3.amazonaws.com/i/owt8a0ptc68nzkntehue.png)

Data is now perfectly balanced.

* 1. Feature Selection

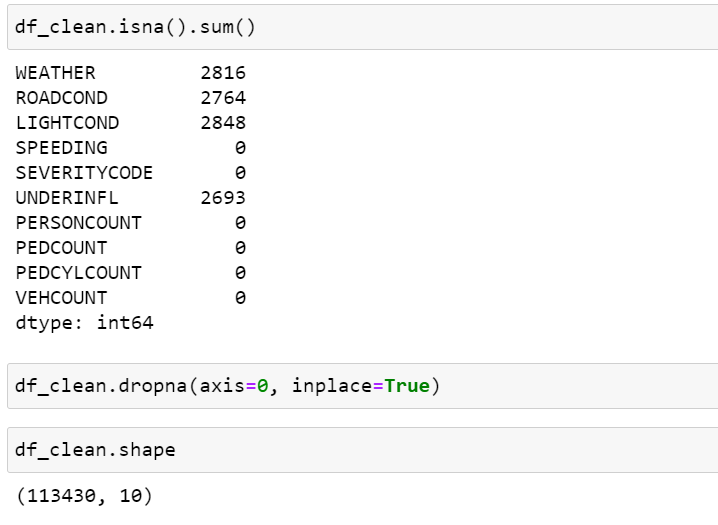
As the dataset has possibly been sourced from a database table, several unique identifiers and spatial features are present in the database which may be irrelevant in further statistical analysis. These fields are OBJECTID, INCKEY, COLDETKEY, INTKEY, SEGLANEKEY, CROSSWALKKEY, and REPORTNO. Other fields such as EXCEPTRSNCODE, SDOT\_COLCODE, SDOTCOLNUM and LOCATION and their corresponding descriptions (if any) are categorical but have a large number of distinct values that shall not be that much useful for analysis. The INCDATE and INCDTTM denote the date and the time of the incident but may not be of use in further analyses. The data needs to be pre-processed.

After studying the features, we finalized the below features to be used for creating our machine learning model.



* 1. Data Cleaning

The dataset consists of many null values that needs to be removed from the dataset in order to create the ML model.



### 2.5. Data type conversions

### Some columns have data types as String which can’t be fed to our ML model directly. Thus, we need to apply one-hot encoding in order to create the model.

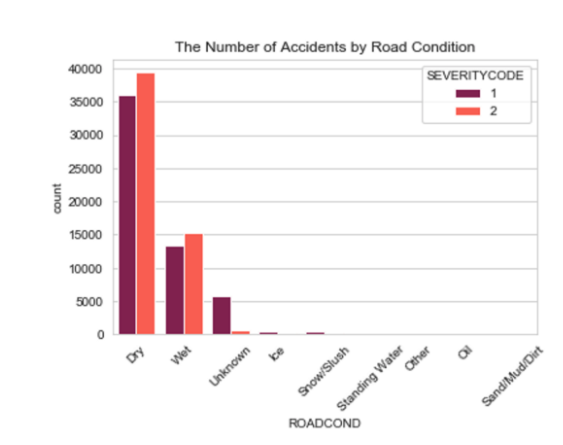
### Out of the available columns, we have to do one-hot encoding of the WEATHER, ROADCOND, and LIGHTCOND fields as they are categorical.

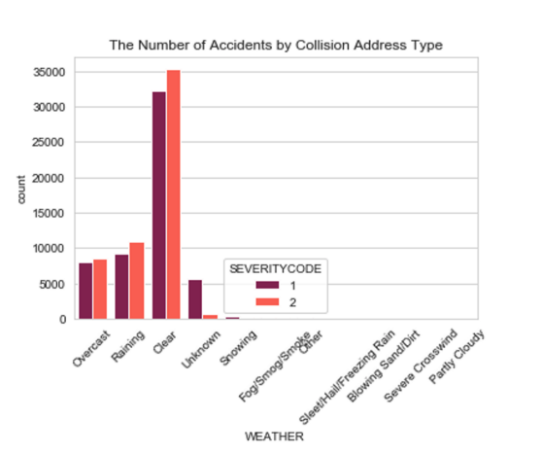
### 

After one-hot encoding, we now have 36 variables in total.

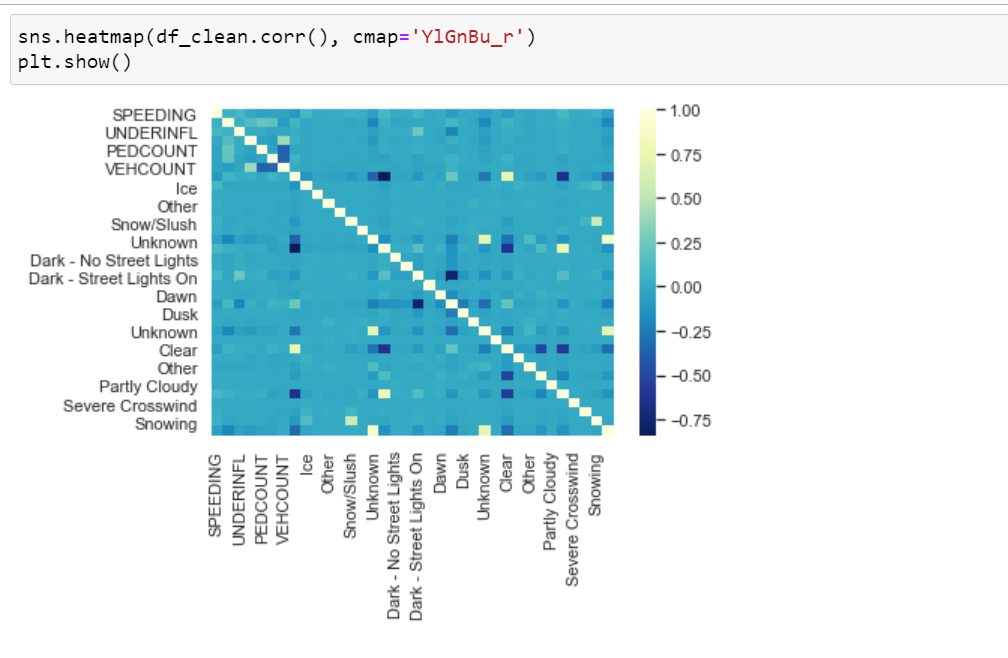
1. Methodology
   1. Exploratory Data Analysis

As per analysis, ROADCOND and WEATHER have a big impact on severity of accidents.





Furthermore, Correlation is a statistical technique that can show whether and how strongly pairs of variables are related. Finding the correlation among the features of the dataset helps understand the data better. For example, in the heatmap shown below, it can be observed that some features have a strong positive / negative correlation while most of them have weak / no correlation.



* 1. Machine learning model

The objective is to create a model which can predict whether the accident is severe or not. Thus, it is simply a classification problem. For the classification model, we will be using the below models and will check the accuracy, f1 score and log-loss to evaluate the model.

1. KNN

KNN will help us predict the severity code of an outcome by finding the most similar to data point within k distance.

1. Decision Tree

Decision Tree makes decision with tree-like model. It splits the sample into two or more homogenous sets (leaves) based on the most significant differentiators in the input variables. To choose a differentiator (predictor), the algorithm considers all features and does a binary split on them (for categorical data, split by category; for continuous, pick a cut-off threshold). It will then choose the one with the least cost (i.e. highest accuracy), and repeats recursively, until it successfully splits the data in all leaves (or reaches the maximum depth).

Information gain for a decision tree classifier can be calculated either using the Gini Index measure or the Entropy measure, whichever gives a greater gain. A hyper parameter Decision Tree Classifier was used to decide which tree to use, DTC using entropy had greater information gain; hence it was used for this classification problem.

1. Logistic Regression

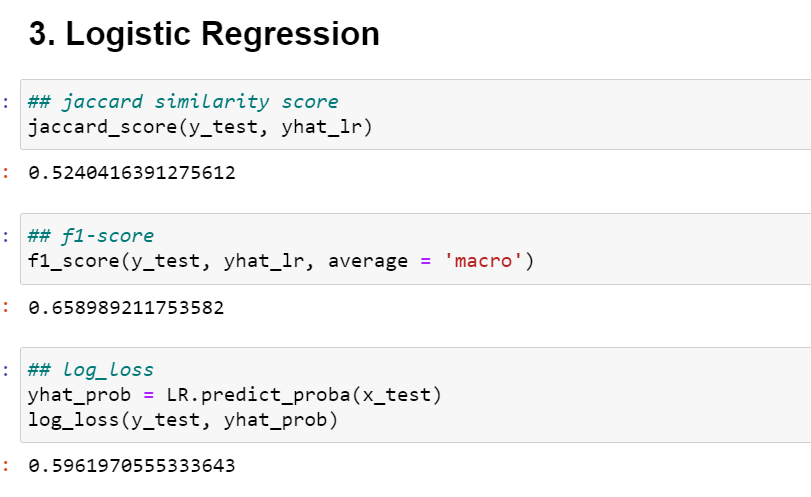
Logistic Regression is a classifier that estimates discrete values (binary values like 0/1, yes/no, true/false) based on a given set of an independent variables. It basically predicts the probability of occurrence of an event by fitting data to a logistic function. Hence it is also known as logistic regression. The values obtained would always lie within 0 and 1 since it predicts the probability.

The chosen dataset has more than two target categories in terms of the accident severity code assigned, one-vs-one (OvO) strategy is employed.

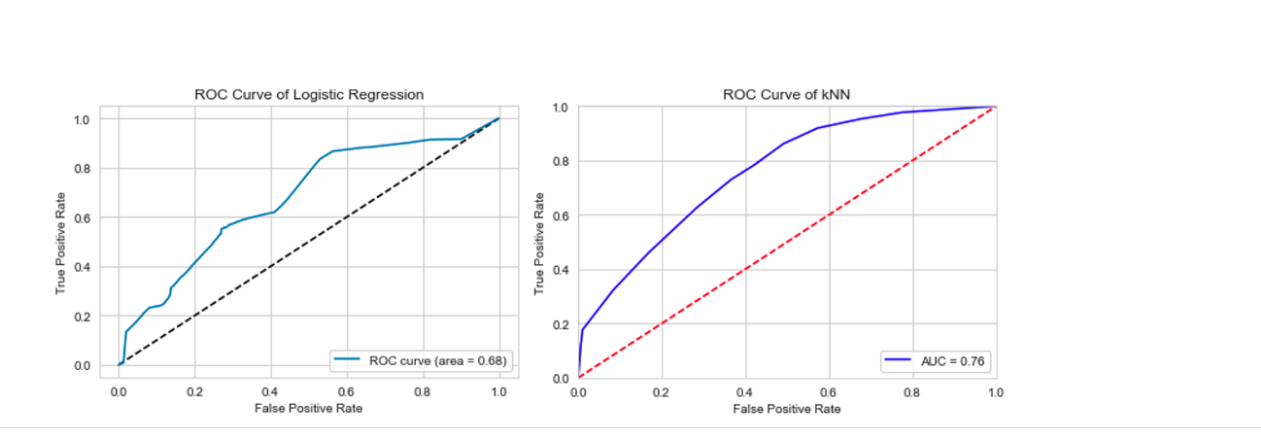
1. Results

After evaluation of all the 3 models, below are the results





ROC Curves:



The above graph shows that the ROC Curve area is better for kNN.

1. Conclusion

In the beginning of this notebook, we had categorical data that was of type 'object'. This is not a data type that we could have fed through an algorithm, so label encoding was used to created new classes that were of type int8; a numerical data type.

After solving that issue, we were presented with another -imbalanced data. As mentioned earlier, class 1 was nearly three times larger than class 2. The solution to this was downsampling the majority class with sklearn’s resample tool. We downsampled to match the minority class exactly with 58188 values each.

Once we analyzed and cleaned the data, it was then fed through three ML models; K-Nearest Neighbor, Decision Tree and Logistic Regression. Although the first two are ideal for this project, logistic regression made most sense because of its binary nature.

Based on historical data from weather conditions pointing to certain classes, we can conclude that particular weather conditions have a somewhat impact on whether or not travel could result in property damage (class 1) or injury (class 2).