**Introduction**

**1.1 Background**

An accident is when a vehicle collides with another vehicle, pedestrian, animal, obstacles, etc. Many factors contribute to collision including road environment, driving skills, speeding, impaired due to drug/alcohol, etc. The total number of persons and vehicles involved, pedestrians and injuries inflicted affect the severity of the accident. Many lives are lost worldwide just in road accidents.

**1.2 Problem**

The number of the accidents all around the world is on a continuous rise. A slight delay in the availability of the medical services may be the difference between life and death. This gap can sometimes be a result of the lack of knowledge of the severity of the accident. Having a clear picture of the accident can help authorities provide better emergency services. A model that can predict the severity by weighting in the different factors of weather, road, visibility, light conditions, etc. can be very effective to bridge the gap. Lot of the accidents are also the result of careless driving. How do the environmental factors impact, and their contributions? Taking all the factors into account is a necessity.

**1.3 Interest**

Following problem is of the interest of city authorities, department of transportation and people in healthcare. The predictability of the severity of an accident will help emergency staff to prepare well for the injured. It can define variables like how many ambulances, emergency medical staff and police are required on the site of accident. This can be a crucial information and decrease the time delay and save lives. City authorities and department of transportation can plan from patterns of weather and road environments and flag warning to people. The ultimate benefit will be to the local public who would receive better services and there can be probable decline in accidents due to warnings.

**Data**

The data required here is one which contains details about the environmental factors as well as the driver details like if the person was under some alcohol/ drug influence. Environmental factors can even lead to a pre-assessment of the probability of travelling unfriendly conditions and warning can be flagged in advance. The data will be used for classification using Logistic Regression and predict the probabilities of the severity of accidents. We will use a historical dataset that fulfils all the requirements. The dataset for this project is provided by Coursera and is available to be downloaded from <https://s3.us.cloud-object-storage.appdomain.cloud/cf-courses-data/CognitiveClass/DP0701EN/version-2/Data-Collisions.csv>. The metadata for the dataset is available at <https://s3.us.cloud-object-storage.appdomain.cloud/cf-courses-data/CognitiveClass/DP0701EN/version-2/Metadata.pdf>. The metadata information was very useful in understanding the dataset well.

We have a total of 194673 rows and 38 columns in our dataset. There are a few unique id columns which will not be an optimum attribute for a model. The dataset has attributes including, speeding, road environments like visibility, light conditions and careless driving like inattention and driving under influence. Many attributes are unique identifiers for accident used by department of transportation and other organizations. A few of the identifiers or different ‘key’ columns had a description column describing the value. Thus, adding the redundant information. The target column ‘severitycode’ had two more columns with the redundant information. Therefore columns ‘severitycode.1’ – an identical column and ‘severitydesc’ were dropped. These attributes are dropped as they will help predict the severity of the accident.

Most of the attributes required close observation and data cleaning. There are columns with as high as 80 to 97% of null values. We will have to impute, drop or work around them. Most of these were in the columns with “Yes” or “NO” values wherein only one of the values was available and the other was left out. These were easily replaced. For example, “speeding” attribute consisted of only ‘Y’ and ‘Nan’ values. Here, it can be inferred that only accident that involved speeding were documented as ‘Y’ and rest were left empty. The value “Nan” can be easily replaced as “N” for No or both “Y” and “Nan” can be replaced by “1” and “0” respectively.

The columns ‘personcount’, ‘pedcount’, ‘vehcount’, ‘roadcond’, ‘weather’, ‘speeding’, ‘underinfl’, ‘inattentionind’ are of keen interest and would be good predictors of severity. More vehicles and people involved tends cause a higher severity accident.

**Methodology**

For the data cleaning, we dropped redundant data, imputed and dropped null values, merged values of categorical variables with very low frequency and one-hot encoded categorical variable. We have a data distribution of 7:3 for severity-code 1 and 2. For the data preparation, the train\_test\_split() module from sklearn is used to split the data into train and test sets. Since, we have an imbalance in the data, it is necessary to define the parameter ‘stratify’. Here it is set to ‘y’, the target value. It ensures that the test set represents the data distribution in the original data. The StandardScaler() module of sklearn library was used to standardize the data. It removes the mean and scales to unit variance.

Once the data is prepared, we are ready for the initial model assessments. In the first test, we determined if the solvers parameter in LogisticRegression had any impact on the output metrics. All the models performed similar with little to no difference. The data was next tested using LogisticRegressionCV. Using Cross-validation and different values of Cs (inverse of regularization strength), we had similar results. The results of above tests were important to decide on a metric of evaluation of models. Our data has 2 target values 1 and 2 with imbalanced distribution. Accuracy is not a good measure here. Having a dummy model predict all samples as 1 will give us an accuracy of 70%. Precision in the ratio of the actual true positives out of the total positives predicted. Recall is the no of true positives predicted out of the total positives in the data. We need to have a high recall of both values with high precision. There we have used ‘f1-score’ as a metric in all models. F1 score is the balance between precision and recall. We require a high f1-score for both classes and therefore metric ‘f1\_macro’ is used. ‘f1\_macro’ is the macro average of the f1-scores of all classes in the target variable. ‘Classification\_report()’ is used to print the various precision, recall and f1-scores of the model prediction and ‘f1\_macro’ is used as the ‘scoring’ parameter.

Using the ‘class\_weight’ parameter as ‘balanced’ in LogisticRegression, we receive a better recall and f1-score with dip in precision for value 2 whereas a dip in recall and f1-score but increase in precision for value 1. The DecisionTreeClassifier model with the default parameters didn’t had any better results. It had a high f1-score for value 1 but a low f1-score for value 2.

After reviewing Decision Tree Classifier and Random forest Classifier, I have decided to work on the imbalance in the data. For this, we will be using the imblearn library. It is used to oversample the minority class and under-sample the majority class. We have used the RandomOverSampler, SMOTE and ADASYN. RandomOverSampler randomly replicates the minority class examples. It is known to increase the likelihood of overfitting. To avoid overfitting, Synthetic Minority Oversampling Technique (SMOTE) is used. It works well in many applications. It generates synthetic data based on feature space similarities between minority instances. Adaptive Synthetic Sampling (ADASYN) generates samples of minority class based on their density distributions. More Synthetic data is generated for the minority samples that are difficult to learn. Resampling best works with the combination of under-sampling of majority and over-sampling of minority class. A combined resampler is available in imblearn – SMOTEENN.

**Results**

After analysing various models using different data resampling techniques, we know that the imbalance in the data had quite an impact on the prediction and the techniques used. LogisticRegression turned out to be a better overall model than RandomForestClassifier. The simplest way to build a model was to include the ‘class\_weight’ parameter as ‘balanced’. After using the resampling techniques from imblearn library, we found that ADASYN and RandomOverSampler turned out to be the best with almost similar results. The combination of under-sampling of majority and oversampling of minority classes did not provide very good results with loss of information. As a lot of samples from majority were dropped. It did not provide a good representation of the actual data. In the analysis, the cost to balance the prediction of the two values also played a crucial role. As the incorrect prediction of value 2 severity has more cost than the incorrect prediction of value 1 severity. Value 2 is considered to be a more severe accident than value 1 (As provided in the metadata). Finally, the logistic regression model trained on the scaled and ADASYN resampled data proved to be the best model. It has an f1 macro score of 0.64 with a value 2 recall of 0.85, a macro recall of 0.71 and macro precision of 0.68. Arguably random over sampled data logistic regression was almost similar to ADASYN model and it had the highest roc\_auc score. But random over sampler is prone to over-fitting data.

**Conclusion**

It was a great learning exercise working on this project. It is open to more possibilities tweaking with data and exploring more data sampling techniques. I plan to work on this dataset more and explore what remains to be learned specially about the imblearn, data imbalance and the reproducibility of models, working on data exploration and insights. As well to learn about accident severity and the various factors leading to it. The impact that such models have on society and safety is huge with large implications. The project is open to more possibilities that I intend to work on.