COURSERA CAPSTONE BLOG – Eknath Reddy

INTRODUCTION/BUSINESS PROBLEM FOR THE PROJECT

In today's world, we travel atleast once a day to reach a certain destination, be it a school,college,office or anywhere. This shows how regularly people commute to differnet places. Though there are many rules and protocols to be followed while travelling on a road, there are always some accidents. These accidents have a serious impact on people's lives.

Hence, using different Data Science techniques on the data available, we can analyse the accidents and predict the severity of each accident in a such a way that necessary precautions can be taken by the City Road Department and the common folk.

The main focus of this project is to predict the severity of an road accident. The predictions and results gained through this capstone project can be used by the Government in order to find out ways to keep the number of accidents to the minimum. Hence, a clear, accurate and detailed analysis will be the goal of my project.

UNDERSTANDING THE DATA INVOLVED

The Dataset being used in this project is from the Seattle Traffic Department, which is based on the collisons occured in the city of Seattle.

The Data is recorded from 2004 to May-2020

The Dataset includes 194,673 samples and 37 attributes

The attributes included are Severity of collision, Weather Condition, No. of People involved, Road Condition, Location, Report number etc.There are also many empty entries in the Dataset.

In the next phase of the project, which is Data Cleaning, all the irrelevant data columns will be dropped and the relevant ones will be modified in way that would benefit the model.

DATA ANALYSIS

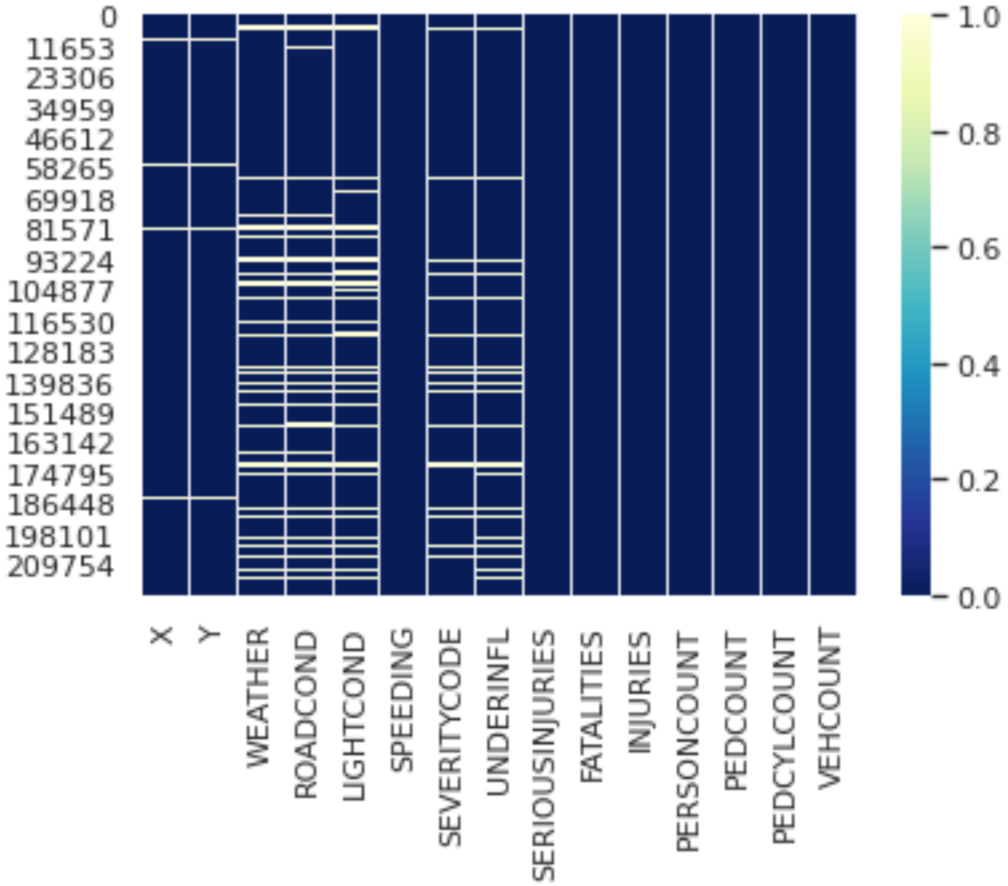
The **`WEATHER`** field contains a description of the weather conditions during the time of the collision. The **`ROADCOND`** field describes the condition of the road during the collision. The **`LIGHTCOND`** field describes the light conditions during the collision. The **`SPEEDING`** field classifies collisions based on whether or not speeding was a factor in the collision. Blanks indicate cases where the vehicle was not speeding.

The **`SEVERITYCODE`** field contains a code that corresponds to the severity of the collision. and **`SEVERITYDESC`** contains a detailed description of the severity of the collision. We can conclude that there were 349 collisions that resulted in at least one fatality, and 3,102 collisions that resulted in serious injuries. The following table lists the meaning of each of the codes used in the **`SEVERITYCODE`** field:

SEVERITYCODE Value	**Description**
1	Accidents resulting in property damage
2	Accidents resulting in injuries
2b	Accidents resulting in serious injuries
3	Accidents resulting in fatalities
0	Data Unavailable i.e. Blanks

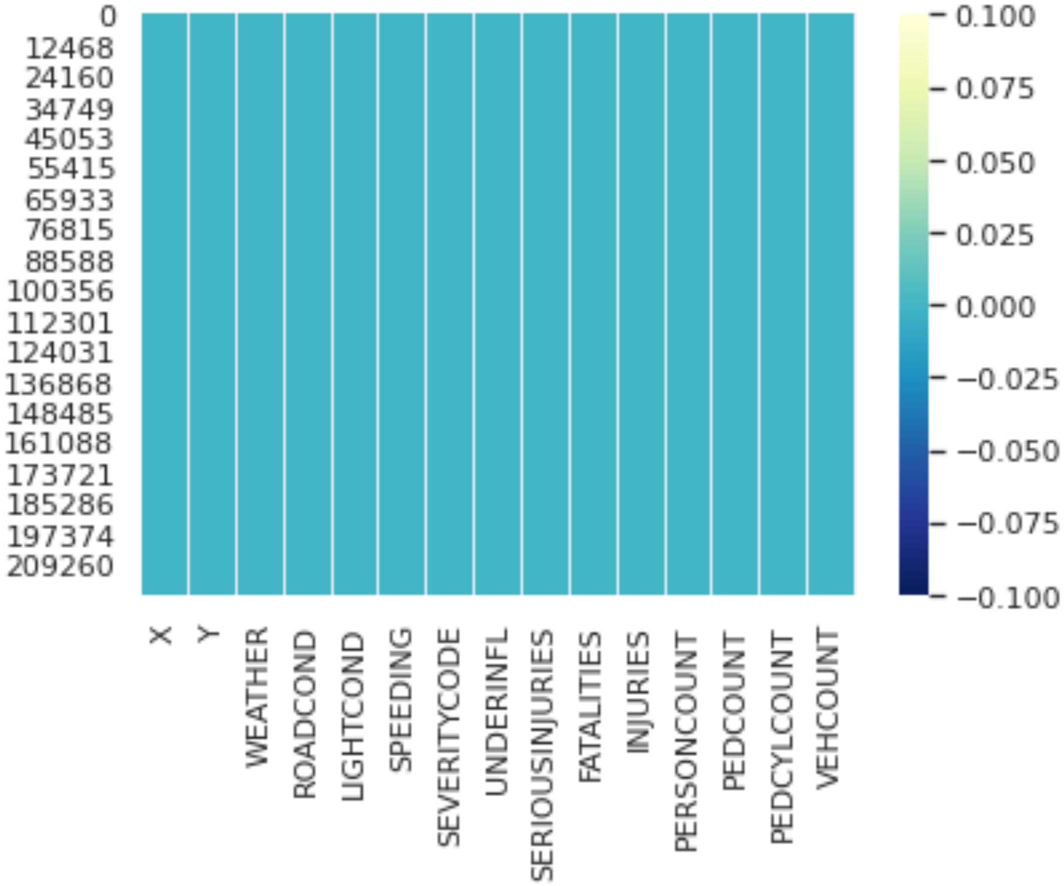
The `UNDERINFL` field describes whether or not a driver involved was under the influence of drugs or alcohol. The values `0` and `N` denote that the driver was not under any influence while `1` and `Y` that they were. The `PERSONCOUNT` and `VEHCOUNT` indicate how many people and vehicles were involved in a collision respectively.

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 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



Dataset before pre-processing

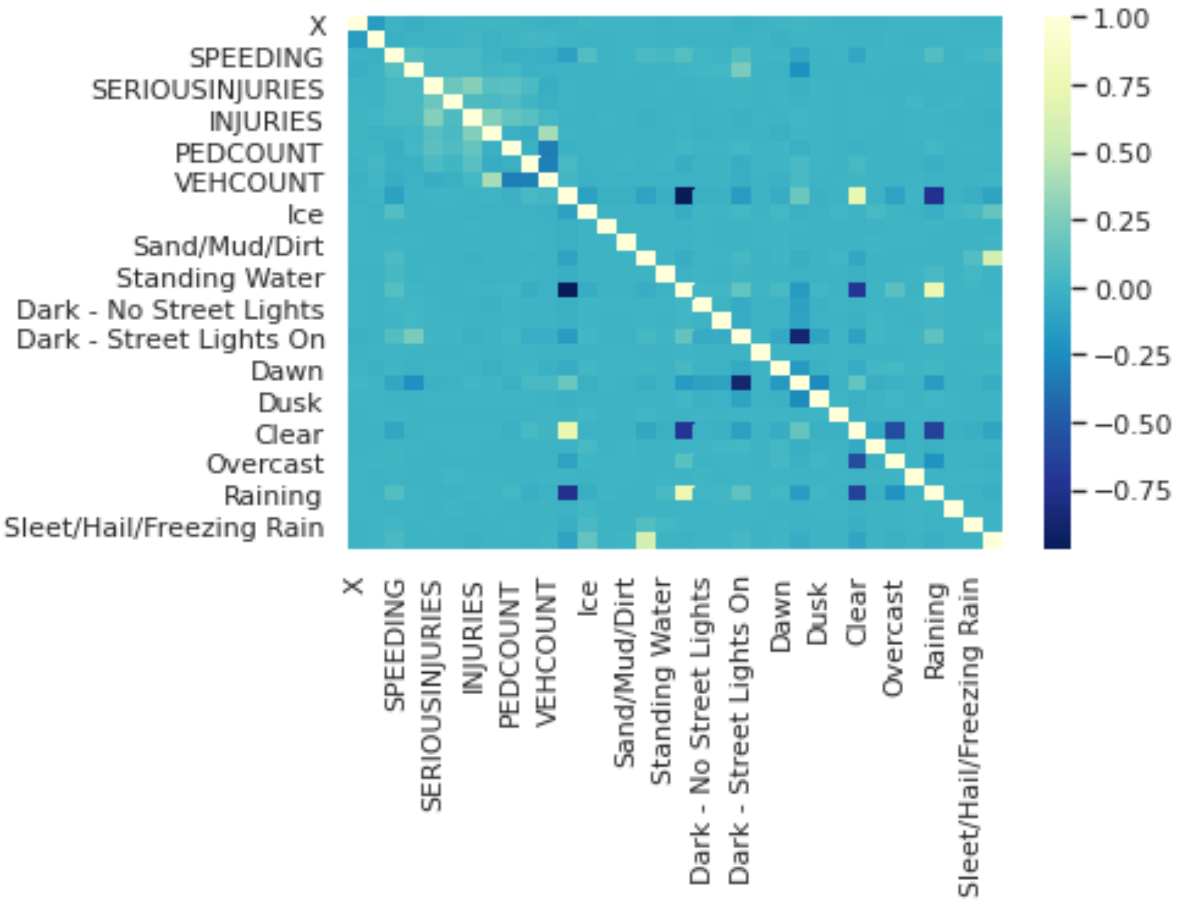
After dropping irrelevant columns and null values and performing data cleaning, we got a dataset with 171,380 rows.



Dataset after data cleaning

After fixing other data inconsistencies, we now do an one-hot encoding of the **`WEATHER`**, **`ROADCOND`**, and **`LIGHTCOND`** fields as they are categorical. Shuffling of the dataset is also necessary as it is an unbalanced dataset.

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.



Correlation Matrix

The datasets x and y are constructed. The set x contains all the training examples and y contains all the labels. Feature scaling of data is done to normalize the data in a dataset to a specific range.

After normalization, they are split into **`x_train`**, **`y_train`**, **`x_test`**, and **`y_test`**. The first two sets shall be used for training and the last two shall be used for testing. Upon choosing a suitable split ratio, 80% of data is used for training and 20% of is used for testing.

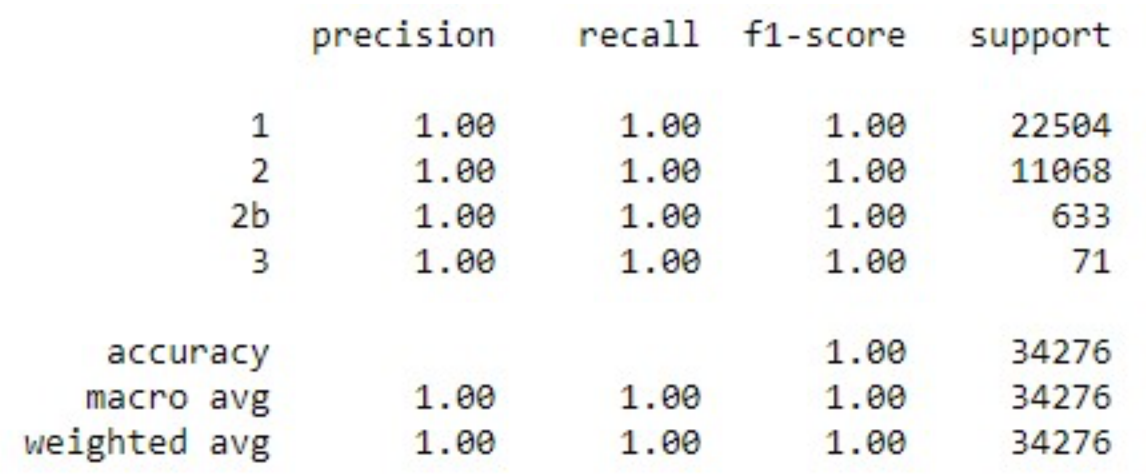
Modelling and Evaluation

Decision Tree Classifier

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.



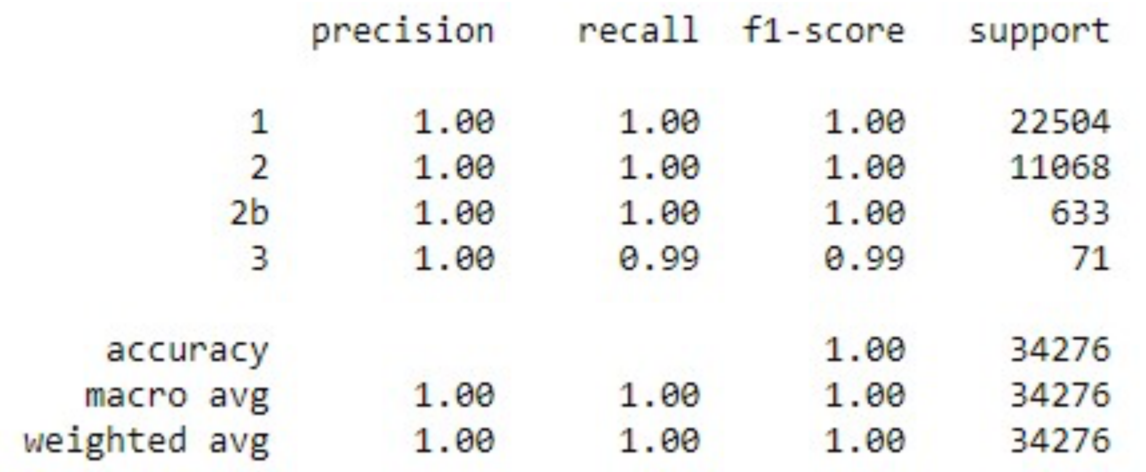
```
              precision    recall  f1-score   support

           1       1.00      1.00      1.00     22504
           2       1.00      1.00      1.00     11068
          2b       1.00      1.00      1.00       633
           3       1.00      1.00      1.00        71

    accuracy                           1.00     34276
   macro avg       1.00      1.00      1.00     34276
weighted avg       1.00      1.00      1.00     34276
```

Classification Report for DTC

Random Forest Classifier

Random Forest Classifier is an ensemble (algorithms which combines more than one algorithms of same or different kind for classifying objects) tree-based learning algorithm. RFC is a set of decision trees from randomly selected subset of training set. It aggregates the votes from different decision trees to decide the final class of the test object. Used for both classification and regression.

Similar to DTC, RFC requires an input that specifies a measure that is to be used for classification, along with that a value for the number of estimators (number of decision trees) is required. A hyperparameter was used to determine the best choices for the above mentioned parameters. RFC using entropy as the measure gave the best accuracy when trained and tested on pre-processed accident severity dataset.

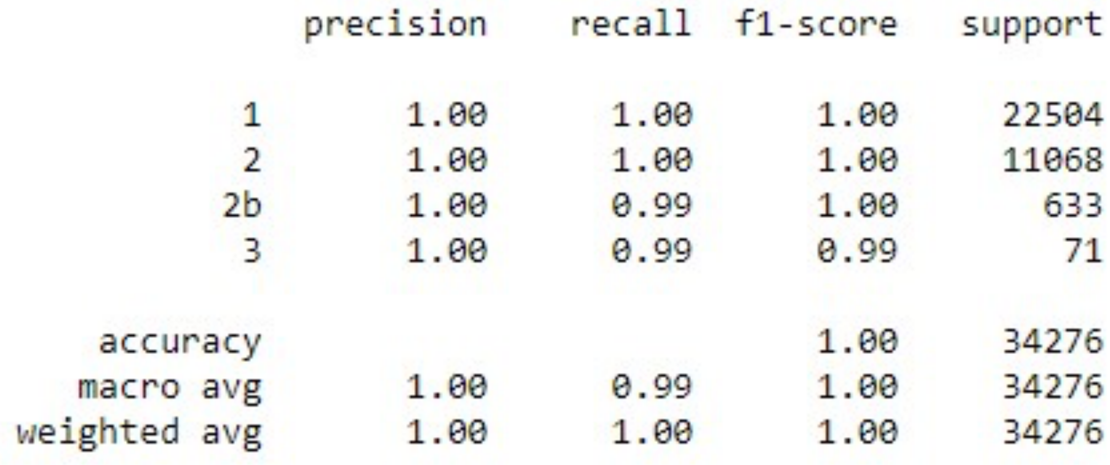


```
              precision    recall  f1-score   support

           1       1.00      1.00      1.00     22504
           2       1.00      1.00      1.00     11068
          2b       1.00      1.00      1.00       633
           3       1.00      0.99      0.99        71

    accuracy                           1.00     34276
   macro avg       1.00      1.00      1.00     34276
weighted avg       1.00      1.00      1.00     34276
```

Classification Report for RFC

Logistic Regression Classifier

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.

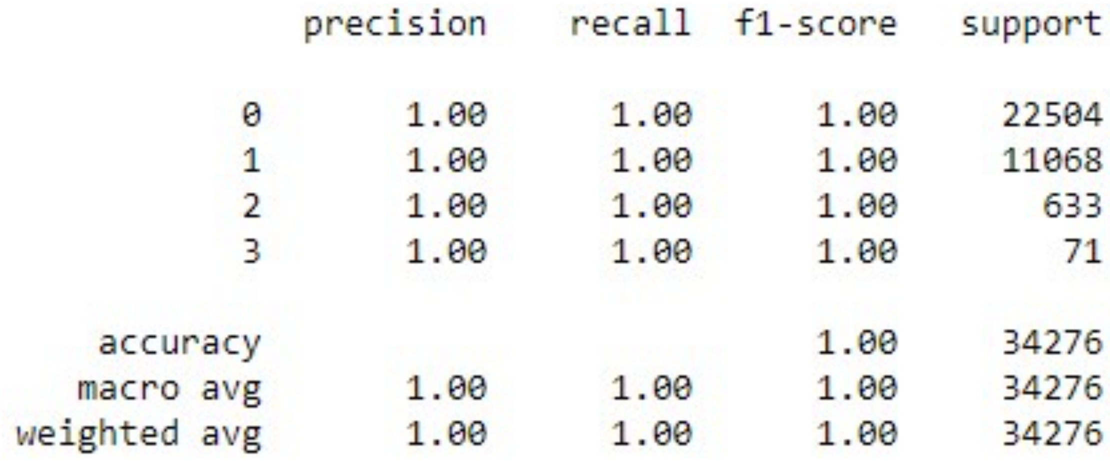


```
              precision    recall  f1-score   support

           1       1.00      1.00      1.00     22504
           2       1.00      1.00      1.00     11068
          2b       1.00      0.99      1.00       633
           3       1.00      0.99      0.99        71

    accuracy                           1.00     34276
   macro avg       1.00      0.99      1.00     34276
weighted avg       1.00      1.00      1.00     34276
```

Classification Report for LogRegClassifier

Neural Network

Neural networks can be used to capture non-linearity between features. We have used a Sequential ANN where there are 4 hidden layers. The `relu` and `sigmoid` activation functions are used. The loss function that is used is `categorical_crossentropy` as the target is integer-coded.

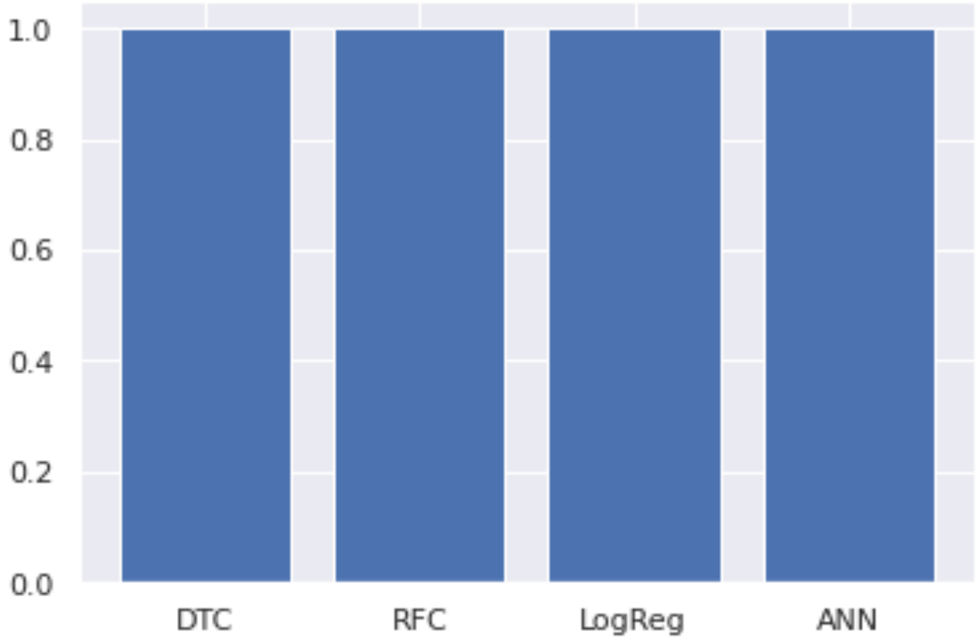


```
              precision    recall  f1-score   support

           0       1.00      1.00      1.00     22504
           1       1.00      1.00      1.00     11068
           2       1.00      1.00      1.00       633
           3       1.00      1.00      1.00        71

    accuracy                           1.00     34276
   macro avg       1.00      1.00      1.00     34276
weighted avg       1.00      1.00      1.00     34276
```

Classification Report for ANN

Results

The accuracies of all models lied was 100% which means we can accurately predict the severity of an accident. A bar plot is plotted below with the bars representing the accuracy of each model.



Conclusion

Initially, the classifiers had an prediction accuracy of 66%-71%, however, upon going back to the data preparation phase, minor tweaking and taking additional fields in the dataset improved the overall accuracy of all models.

The accuracy of the classifiers is excellent, i.e. 100%. This means that the model has trained well and fits the training data and performs well on the testing set as well as the training set. We can conclude that this model can accurately predict the severity of car accidents in Seattle.

Future Work

The trained model can be deployed onto governance and monitoring web and mobile applications to predict the accident severity for a given set of parameters.