Capstone Project: Car Accident Severity

Final Report

1. Problem statement

In this project we look at the severity of car accidents. Severity is grouped in two categories: 1 Property Damage Only Collision and 2 Injury Collision. We believe that municipalities can use this prediction to improve roads and road signs. Drivers can be more cautious due to some specific conditions.

2. Dataset

This data set is collected by SDOT Traffic Management Division, Traffic Records Group. It contains collisions that happened in Seattle from 2004 to present. Data set obtained from a csv file and it has 194673 rows and 38 columns including the target column. Our target column is SEVERITYCODE. SEVERITYCODE has two values 1 and 2. 1 is Property Damage Only Collision and 2 is Injury Collision. So this is a binary classification problem.

Some features are longitude, latitiuture, location, time, date, collision type, weather, road condition, light condition.

3. Cleaning and Feature Selection

First I checked if the data set is balanced or not. This is a slightly unbalanced dataset. Percentage of Severitycode categories are:
1 = 70%
2 = 30%

For now I do not deal with unbalanced problem, I will take care of it in the machine learning process.

Secondly I checked missing values in each column. Data set has approximately 200000 rows and after investigation of missing values I found that some features have a very big portion of

missing values. I dropped features which have more than 75000 missing values. These columns are 'INTKEY', 'EXCEPTRSNCODE', 'EXCEPTRSNDESC', 'INATTENTIONIND', 'PEDROWNOTGRNT', 'SDOTCOLNUM', and 'SPEEDING' are dropped.

I checked missing values in each column again and found that some features have missing values around 5000. This time instead of deleting all columns I only delete rows with missing values.

After cleaning the data set from missing values, the data set has dimensions of 180067, 31. I checked unique values for each feature. Four features have a number of 180067 unique features. These remind me that they are a kind of identification number. Indeed when I look at the description of the features these are identification numbers given to each case. I dropped these columns: ['OBJECTID', 'INCKEY', 'COLDETKEY', 'REPORTNO'].

X, Y and LOCATION columns are also dropped because these columns are related to the location of the accident.

At this point I check unique values again and I see that features are mostly categorical.

4. Exploratory Data Analysis (EDA)

Selected features and their unique values:

SEVERITYCODE Column has number of 2 unique values.
STATUS Column has number of 2 unique values.
ADDRTYPE Column has number of 3 unique values.
SEVERITYCODE.1 Column has number of 2 unique values.
SEVERITYDESC Column has number of 2 unique values.
COLLISIONTYPE Column has number of 10 unique values.
PERSONCOUNT Column has number of 47 unique values.
PEDCOUNT Column has number of 7 unique values.
PEDCYLCOUNT Column has number of 3 unique values.
VEHCOUNT Column has number of 13 unique values.
INCDATE Column has number of 5985 unique values.
INCDTTM Column has number of 162058 unique values.
JUNCTIONTYPE Column has number of 7 unique values.
SDOT_COLCODE Column has number of 39 unique values.
SDOT_COLDESC Column has number of 39 unique values.
UNDERINFL Column has number of 4 unique values.

WEATHER	Column has number of	11 unique values.
ROADCOND	Column has number of	9 unique values.
LIGHTCOND	Column has number of	9 unique values.
ST_COLCODE	Column has number of	115 unique values.
ST_COLDESC	Column has number of	62 unique values.
SEGLANEKEY	Column has number of	1955 unique values.
CROSSWALKKEY	Column has number of	2198 unique values.
HITPARKEDCAR	Column has number of	2 unique values.

At this point the percentage of Severitycode categories are: 1 = 69% and 2 = 31%. The ratio is 2.22. This means if any features' any value shows a different percentage other than 2.22 for severity, it will have separation power.

Let's analyze them visually and statistically.

STATUS feature has no variance(Matched= 180066, Unmatched=1) so I dropped it.
SEVERITYCODE.1 feature is the same as SEVERITYCODE so I dropped it.
SEVERITYDESC feature is the same as SEVERITYCODE so I dropped it.

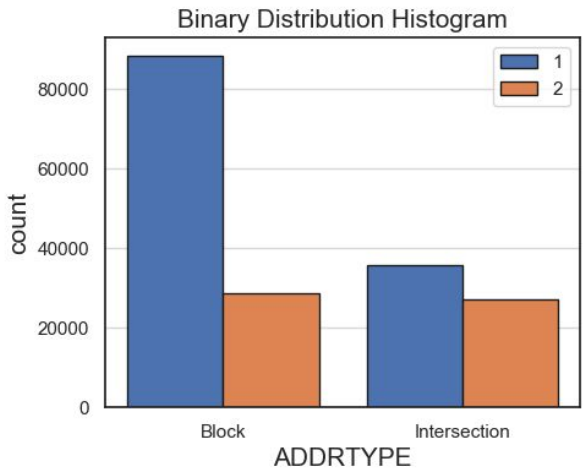

Fig 1: ADDRTYPE

has definitely resolving power.

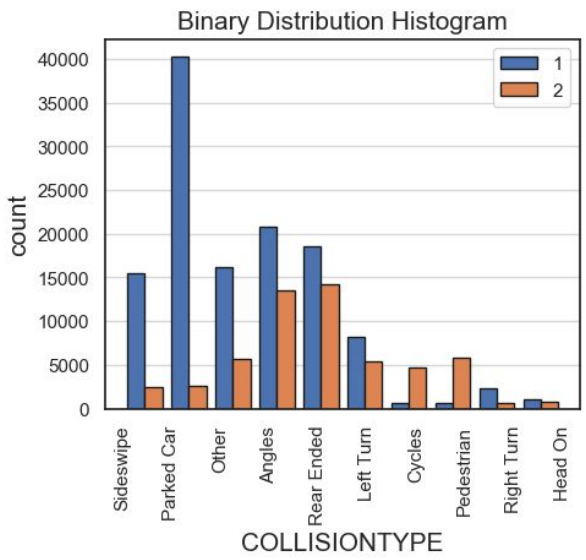

Fig 2: COLLISIONTYPE

has definitely resolving power.

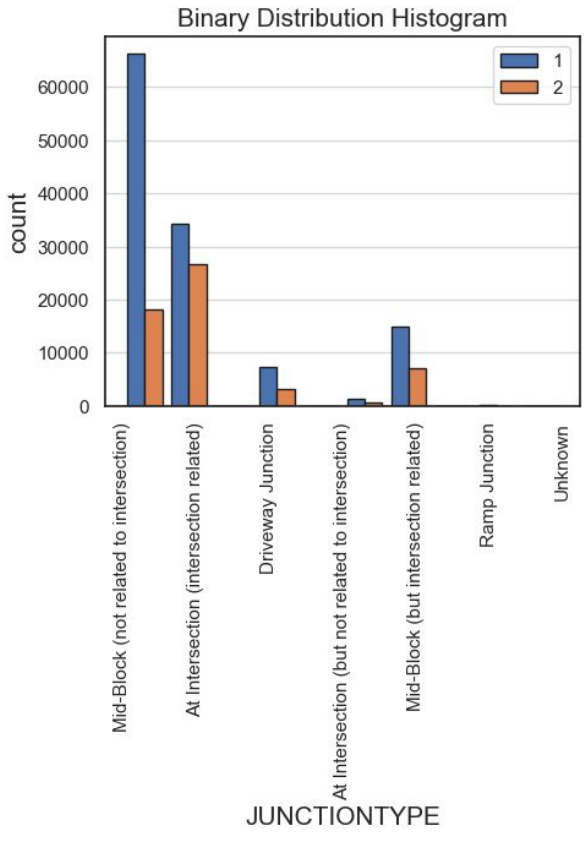

Fig 3: JUNCTIONTYPE

has definitely resolving power.

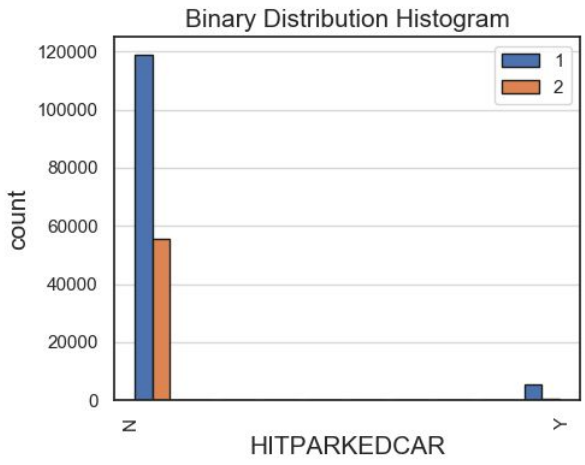

Fig 4: HITPARKEDCAR

has definitely resolving power.

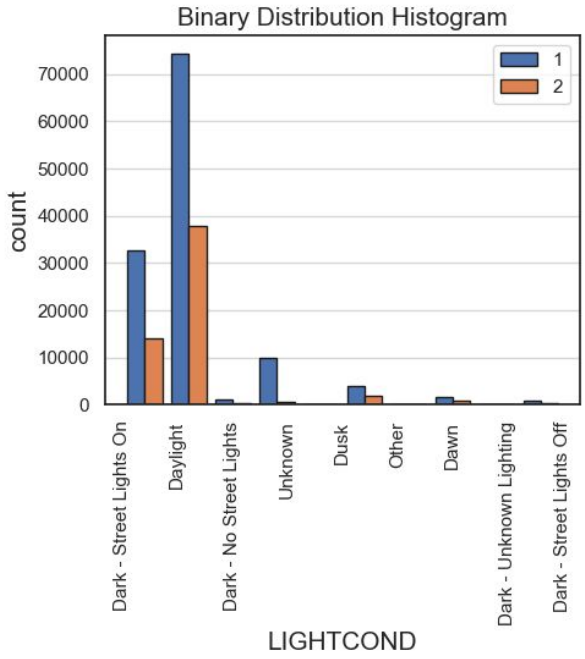

Fig 5: LIGHTCOND

has definitely resolving power.

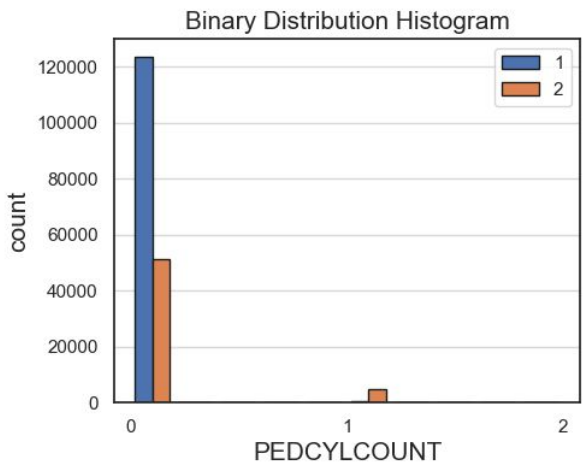

Fig 6: PEDCYLCOUNT

has definitely resolving power.

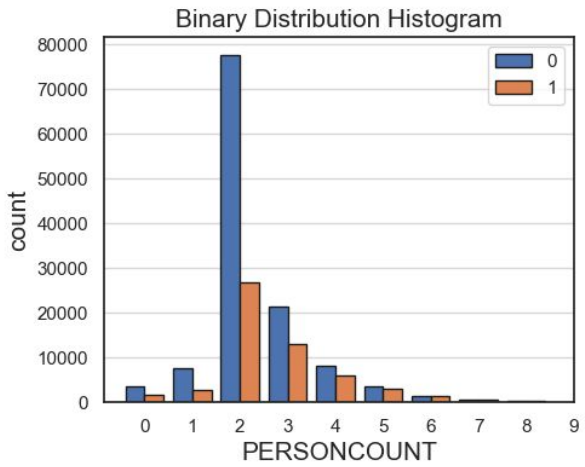

Fig 7: PERSONCOUNT

has definitely resolving power.

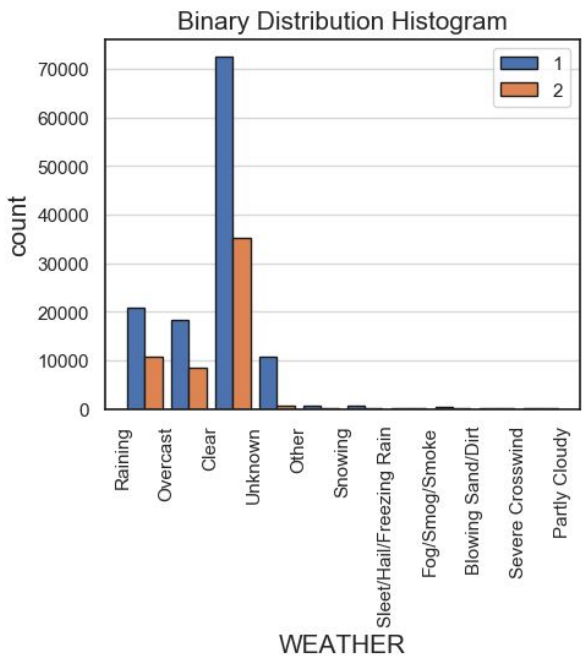

Fig 8: WEATHER

has definitely resolving power.

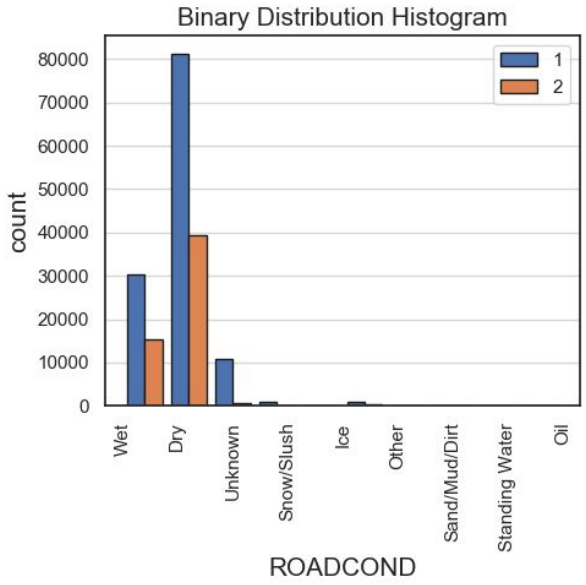

Fig 9: ROADCOND

has definitely resolving power.

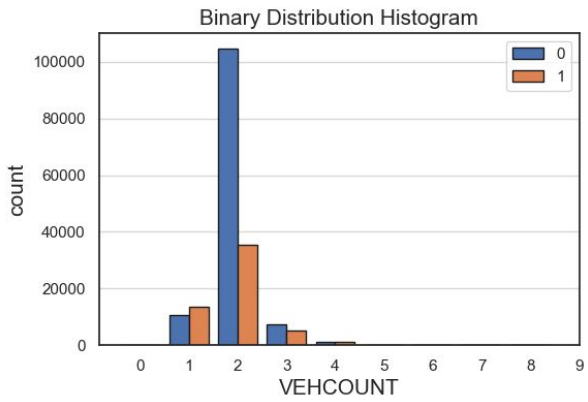

Fig 10: VEHCOUNT

has definitely resolving power.

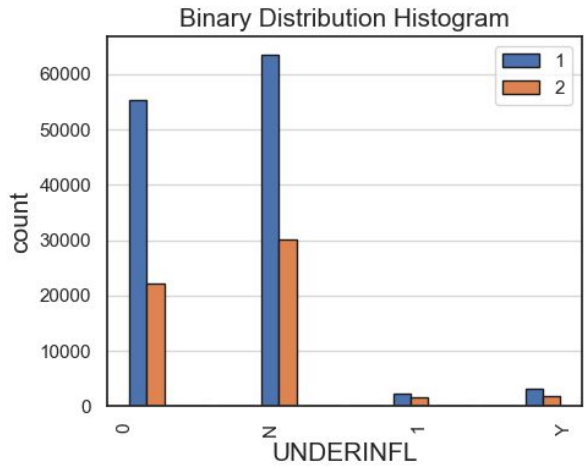

Fig 11: UNDERINFL

has definitely resolving power.

5. Predictive Models

For machine learning I have to do some preparation. I separate features and the target column. X is the dataframe with features and Y is the target.

11 features' datatypes are object type. I convert these object features into categorical values. This is a must for the machine learning process.

ST_COLCODE feature is a combination of integer and string so I converted to numerical values.

I used the labelencoder function for labeling. After creation of new int64 features I dropped object type features.

We have 18 features with all int64 data type.

Using train_test_split function data set is divided into train(X_train, y_train) and test sets (y_train, y_test) with ratio 80:20. Train sets are trained with 3 different machine learning models. Logistic Regression, Random Forest Classifier, and K Neighbours Classifier. The data set is unbalanced, because of this random under sampling (RUS) method is applied. I reported confusion matrices and classification reports for all base and and their RUS models.

I compared weighted average f1-scores, recalls for 1, and AUC.

Model Name	Base Model	RUS Model
Logistic Regression	0.67	0.63
Random Forest Classifier	0.72	0.68
K Neighbours Classifier	0.72	0.70

Table 1: Weighted Average F1-score

Model Name	Base Model	RUS Model
Logistic Regression	0.57	0.64
Random Forest Classifier	0.64	0.69
K Neighbours Classifier	0.65	0.61

Table 1: AUC

I decided the K Neighbours Classifier base model is the best.

5.1. Parameter Tuning for K Neighbours Classifier

After deciding for the best model I did parameter tuning. Grid search is used for parameter tuning. In the grid search I look at the best option from between 1 and 10 numbers of neighbours. After finding the best parameters, the train dataset is trained with those best parameters. The best number of neighbours is found to be 8 in the range of 1 to 10.

After tuning Weighted Average F1-score is decreased to 0.71 and ROC score is decreased to 0.63.

6. Conclusion

Random under sampling was not helpful to my models. But I tried and compared results. Random forest and KNN works better than logistic regression.

I have not included time and day. The next step will be using accident time and day in the machine learning process.