Prediction and Analysis of Traffic Accident in Seattle, Washington

Anugrah Nurhamid

September 05, 2020

1. Introduction

Dataset that we have is about collision mainly on transportation that record from by SPD and will display at the intersection or mid block a segment. So many collisions in this dataset cause that record from 2004 until Present.

We have information about the location, type, when that collisions it happens. This dataset have a condition of the driver, what injuries that they got, and much more about the features that we have.

So, the goals of this project is reduce the accident on transportation by understanding about characteristic each data by severity condition. Finding the insight for each level on severity code, how we can give a guidance when we using means of transportation, how we can more protective when we driving and many more the insight would be suggest in this project.

Mainly this project can be predicted for each level on severity of collisions , for build a better regulations when people using public or private transportation.

2. Data Acquisition and Cleaning

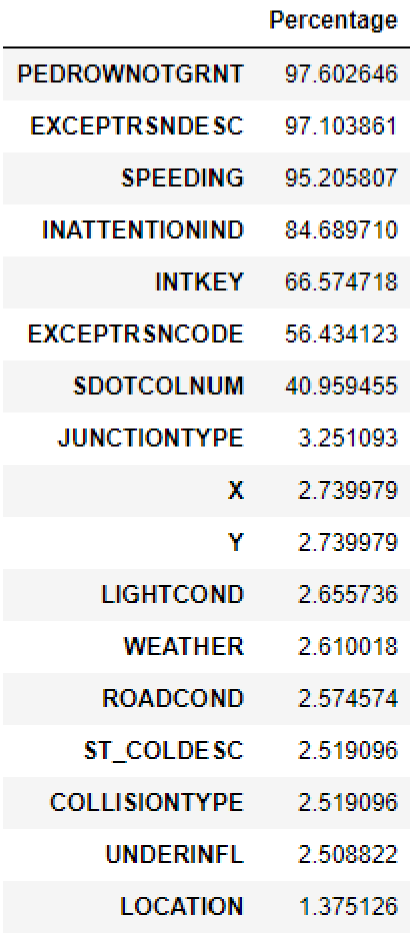
2.1 Data Sources

The data that we have is from IBM Data Capstone Course. That have many attributes (features) for we give a better solution for reduce an accident in Seattle. So, with all features we can take many insight, pattern or maybe we can give a solution in the next modeling. We would to know about the characteristic about the accident that happens in this city. How much features it would be give a high level of severity in that accident. And Finally with all this features we can chosen a great correlation independent variables to dependent variables to give a better machine learning that we build in the final section.

2.2 Data cleaning

From this dataset we have a **target** is **severity code** and the description on column **severity description**. The labeled that we have look's is **imbalanced dataset** dominated by Property Damage is around 70% and another label is People Injury with 30 %.

In this cleaning data, we can try with detection outlier for each columns (if any). Pre-processing the data like inconsistent data or missing value that we have in this dataset.



	Percentage
PEDROWNOTGRNT	97.602646
EXCEPTRSNDESC	97.103861
SPEEDING	95.205807
INATTENTIONIND	84.689710
INTKEY	66.574718
EXCEPTRSNCODE	56.434123
SDOTCOLNUM	40.959455
JUNCTIONTYPE	3.251093
X	2.739979
Y	2.739979
LIGHTCOND	2.655736
WEATHER	2.610018
ROADCOND	2.574574
ST_COLDESC	2.519096
COLLISIONTYPE	2.519096
UNDERINFL	2.508822
LOCATION	1.375126

From table above we can conclude we have many missing value on 7 columns above **(the table has been sorted by percentage)**. We would try to drop all columns that have many missing values and drop rows that have missing value around 3%.

We still have a missing value with percentage around 3 %, We assume we can drop the rows of the missing values, because we still have many data. If this rows has drop we still have 180067 rows data from 194673, its still many data can be explore in visualization and many hypotheses can we build.

In final section, we try to simplify the unique values for each columns, especially columns **Weather, Conditon of Road, Influence of drugs or alcohol, Light Condition**. Many category have simplified by each function above.

- Unknown Category (If no one can describe the condition)
- Other Category (If the condition not included in the existing category)

2.3 Data Visualization

Creating an interactive visualization that lets we dive down into each data point. From data exploration we can start from hypotheses about our data and the our problem need to be tackling. Finally we can take many insight for many solution or many characteristic (detection pattern or something) for each level of severity.

2.4 Feature Engineering

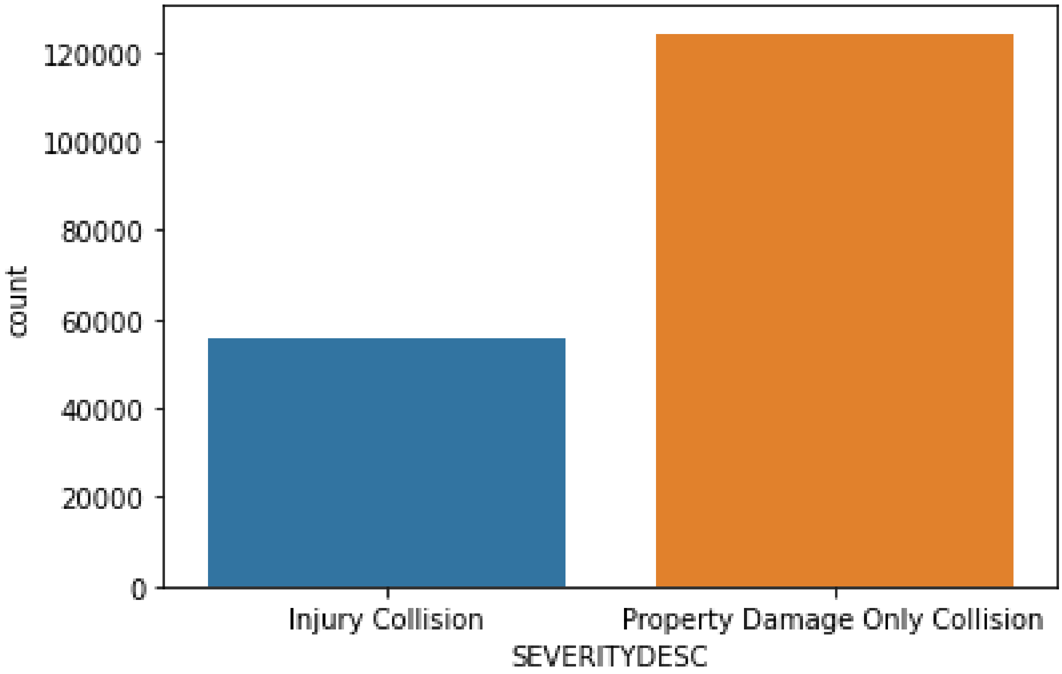
Feature engineering attempts to increase the predictive power of learning algorithms by creating features from raw data that help facilitate the learning process. Beside of future engineering, we can selects the key subset of original data features, its call feature selection

After selection features we have 11 features for next modeling. Before that we must, doing a feature engineering and then split the data into data train and data test. The target that we have is severity describe, so we can describe the prediction is injury or just property damage.

The Feature engineering that we do is one hot encoding, because all categorical features don't have a level like ordinal data such as grade, values something, rating and many more

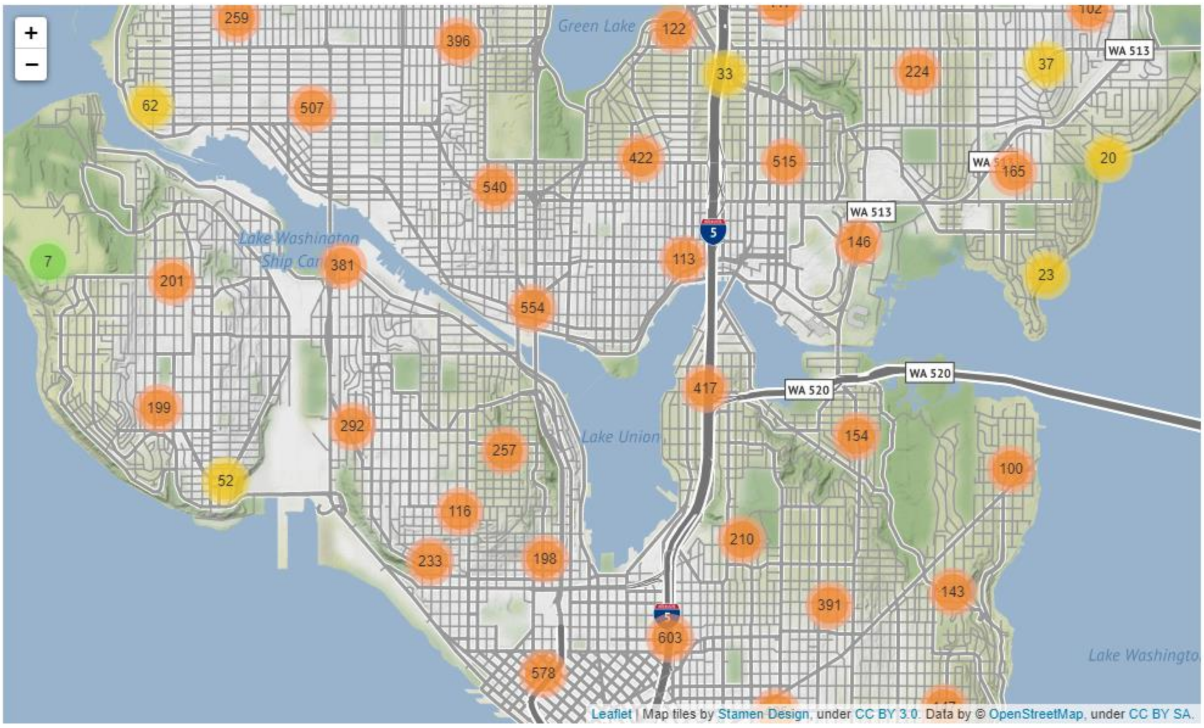
3. Exploratory Data Analysis

Let's explore what insight we can get from this dataset. All columns in exploratory data analysis maybe not included to modeling, but we can more visualize with all columns that we have now.



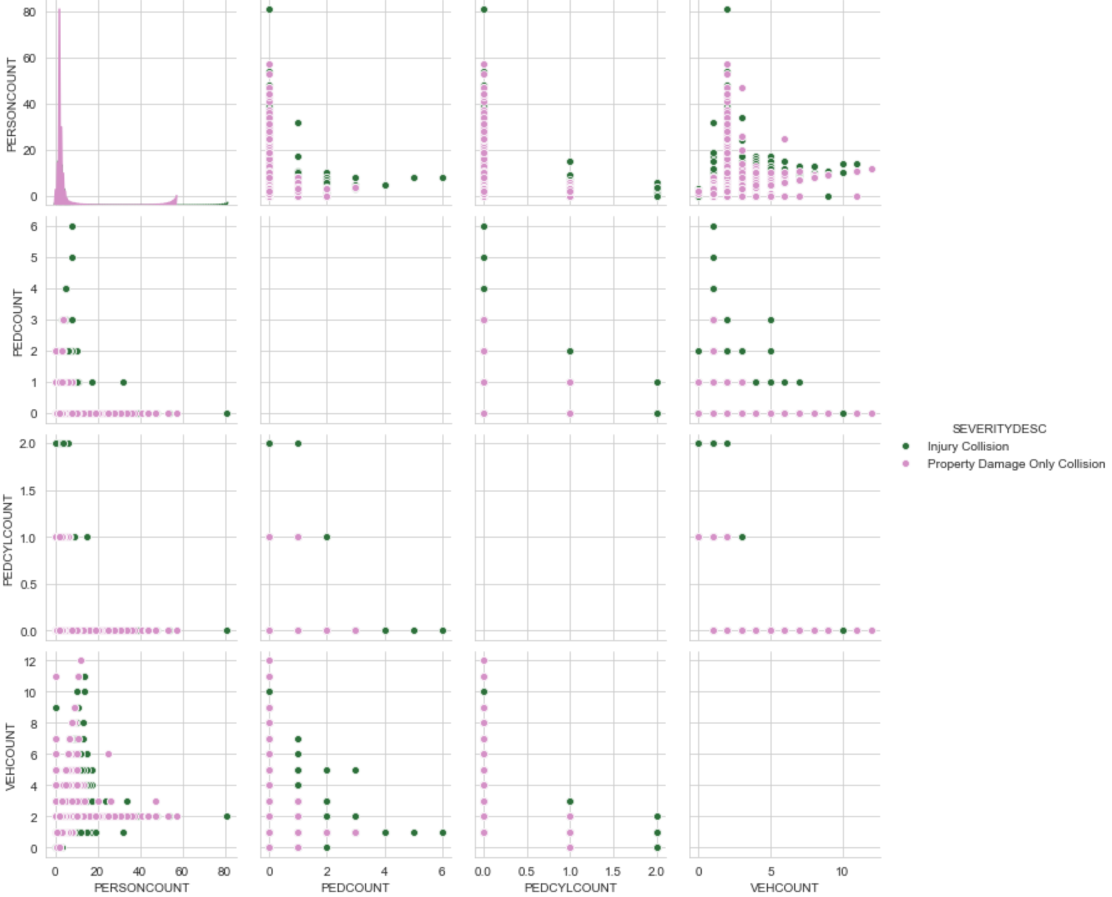
This is the our target we can use *'severity description'* or *'severity code'* to explore the our data. Some visualization in this case we can compare with our target. Because of that, we can take some insight for more deep to learn and understanding some features, what features have high correlation with our target.

3.1 Distribution of Collisions by grouping location



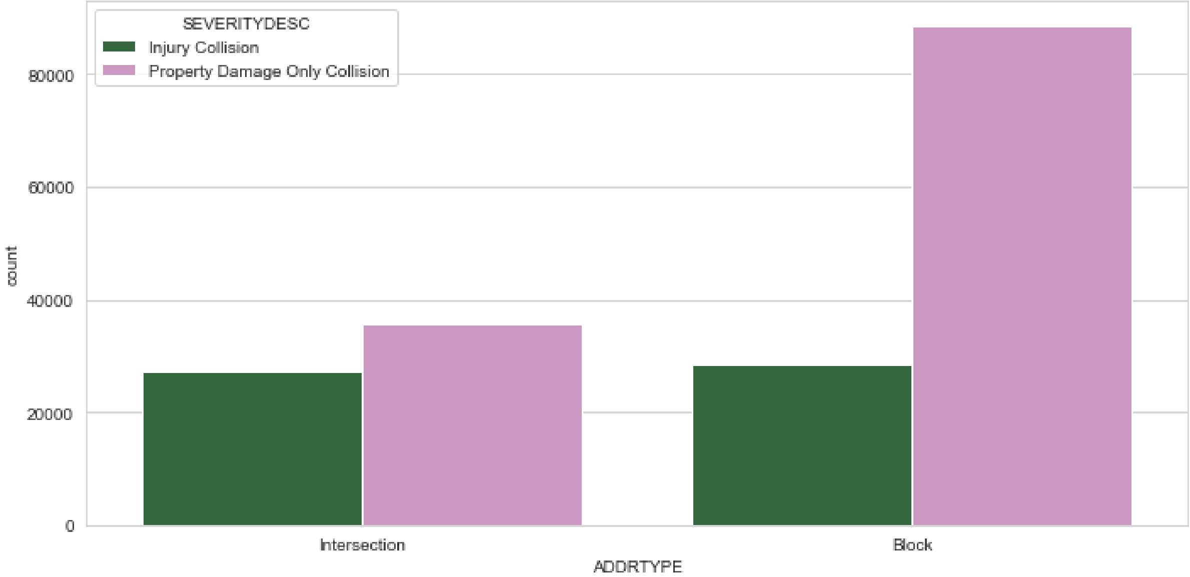
As we see on this folium graph, we grouping for each location on an accident happens. This is a distribution and the total of accident in Seattle.

3.2 Show distribution for numerical Variables

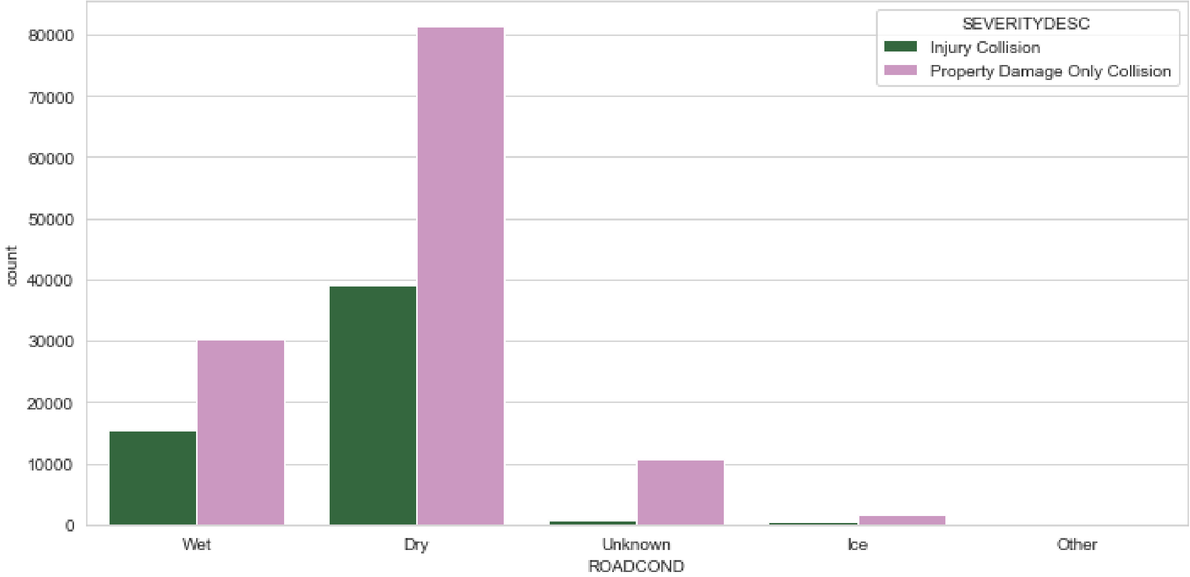


From this graph we conclude, more vehicle count or person count not affected on our target, are the type of accident t is injury collision or just property damage?

3.3 Analysis categorical variables with the target

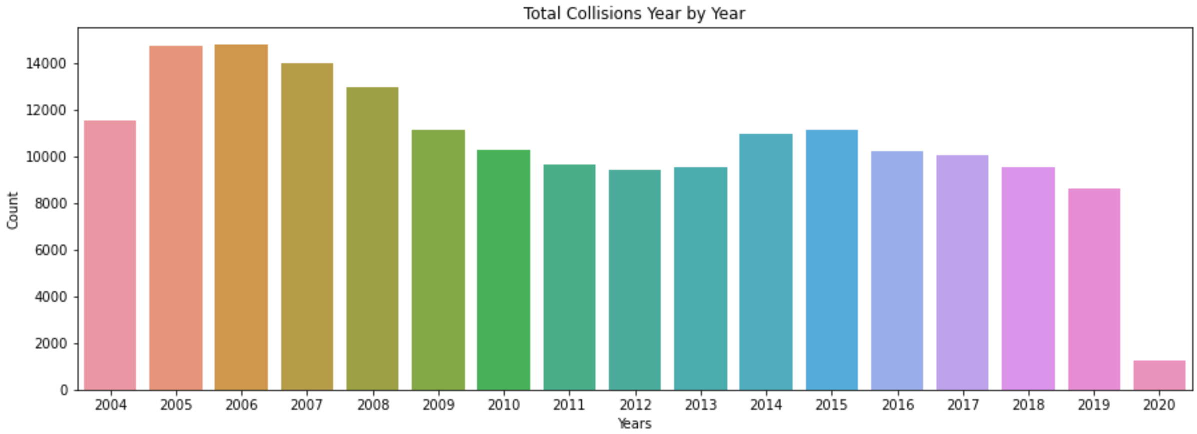


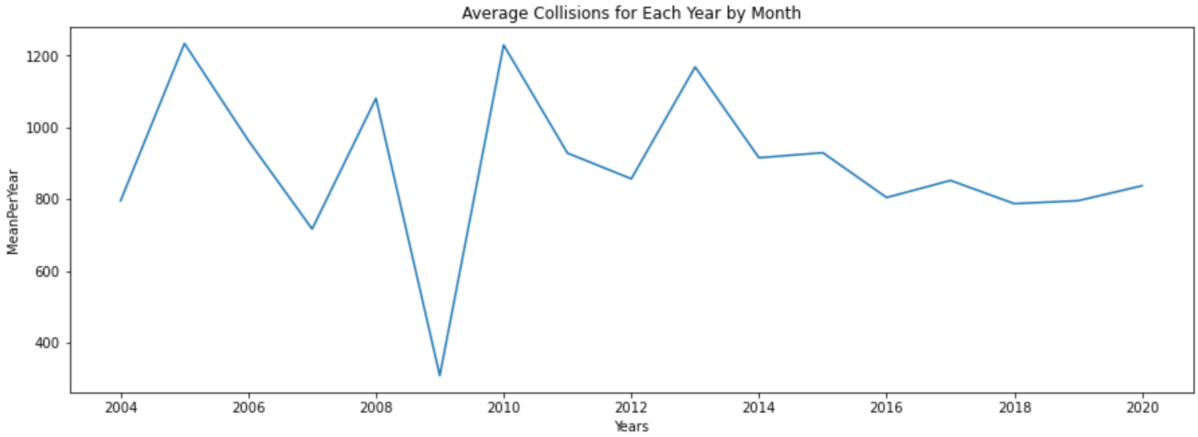
From percentage we know about many people have an accident on address type in block, I assume that block more than dangerous if we know aware about safety riding and the percentage of injury collisions on intersection **44%** more than on block with **24%**.



The condition of road when accident happens dominated by the road condition when wet and dry with around **33 %** of the data injury collisions. ***The conclusion about this categorical features analysis, maybe around 40% percent weather, light condition, condition of road is affected to the accident collisions*****.**

3.4 Analysis collisions year by year





From this 2 graph above we can conclude that.

- Average collisions per month in one year the lowest year is 2009 in range 2004 until 2020 with 300 collisions accident.
- Pattern in total collisions for each year is decreased year by year until 2020.
- The highest total collisions in the data from 2004 increasing to 2005 and hold until 2006 and decrease after that.
- I assume in 2004 until 2006, the Regulations not yet completed and many people many people are not aware of the importance of safe driving. As time goes by until now, the collisions accident is better than the previous year.
- The dataset so many lacks data on 2020, because 2020 not yet completed.

4. Predictive Modeling

In this modeling we wanna try some algorithm is.

- Logistic Regression (optimize parameters and evaluation)
- Random Forest Classifier (Optimize parameters and evaluation)
- Light GBM Classifier (Optimize parameters and evaluation)

As we know the target is **severity code (it's an accident is injury or just property damage)**. I Assume some features have a high correlation with our target to give a better metrics evaluation *(I has tried with correlation ratio in another notebook)*. The metrics would be **optimze is a recall**, after that we see the **accuracy of the model** for predicted an injury collisions or just property damage.

After selection features we have 11 features for next modeling. Before that we must, doing a feature engineering and then split the data into data train and data test. The target that we have is severity describe, so we can describe the prediction is injury or just property damage.

The Feature engineering that we do is one hot encoding, because all categorical features don't have a level like ordinal data such as grade, values something, rating and many more

- Final hyperparameter tuning in Logistic Regression we have **'{'class_weight': None, 'penalty': 'l1', 'solver': 'liblinear'}'**. After that predict with data test and the final result for each metrics in classification report we can describe after this.
- Final hyperparameter tuning in Random Forest Classifier we have **{'max_depth': 4, 'max_features': 2, 'min_samples_leaf': 2, 'min_samples_split': 10}**. After that predict with data test and the final result for each metrics in classification report we can describe after this.
- Final hyperparameter tuning in Light GBM we have **{'boosting_type': 'dart', 'class_weight': {0: 1, 1: 8}, 'num_leaves': 50}**. After that predict with data test and the final result for each metrics in classification report we can describe after this.

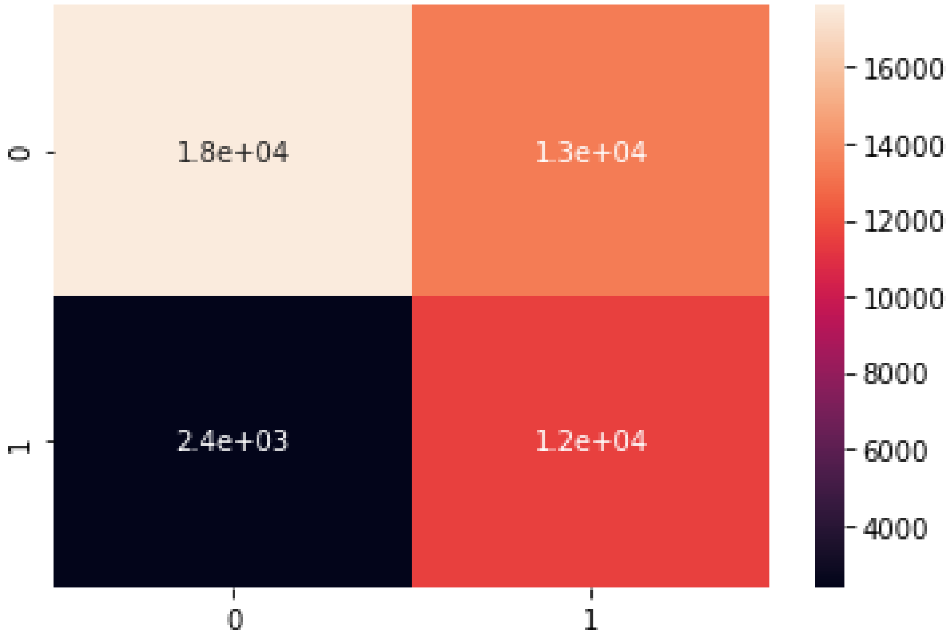
5. Result

From 3 model that we build, we can conclude that the best model and we can take from our conslusion is modeling with Random Forest Classifier. Because, the model have a score **recall 83%** and **the accuracy is 65%**, like we discuss before. In this case we can optimize recall after that we evaluation the best accuracy score. If we compare with **LGBM Model that have recall 96% but the accuracy score is 55 %**. Beside we optimze recall, we compare the accuracy score too, for balancing prediction of the model.

Models that are created still can be optimze.

- Handling the imbalance data with undersampling or oversampling
- More hypterparameter tuning for each model
- Splitting the data with pipeline after using undersampling or over sampling
- Selection features with another method for correlation each features (In this case we try using correlation ratio)
- Evaluation the model with learning curve (The model is overfitting or underfitting)

6. Conclusion



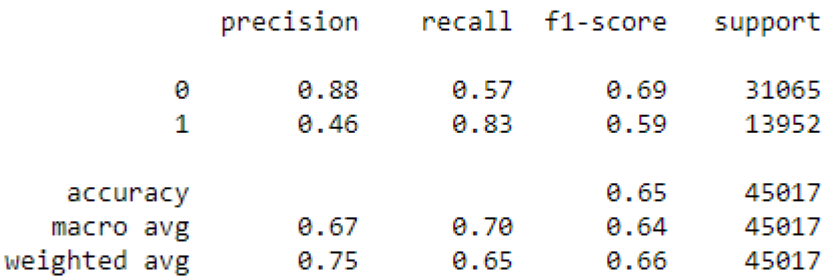
Confusion Matrix

True positive and true negatives are the observations that are correctly predicted and therefore shown in green. We want to minimize false positives and false negatives so they are shown in red color. These terms are a bit confusing. So let's take each term one by one and understand it fully.

- True Positives (TP) - These are the correctly predicted positive values which means that the value of actual class is yes and the value of predicted class is also yes. E.g. if actual class value indicates that this passenger survived and predicted class tells you the same thing.
- True Negatives (TN) - These are the correctly predicted negative values which means that the value of actual class is no and value of predicted class is also no. E.g. if actual class says this passenger did not survive and predicted class tells you the same thing.

False positives and false negatives, these values occur when your actual class contradicts with the predicted class.

- False Positives (FP) – When actual class is no and predicted class is yes. E.g. if actual class says this passenger did not survive but predicted class tells you that this passenger will survive.
- False Negatives (FN) – When actual class is yes but predicted class in no. E.g. if actual class value indicates that this passenger survived and predicted class tells you that passenger will die.



```
              precision    recall  f1-score   support

           0       0.88      0.57      0.69     31065
           1       0.46      0.83      0.59     13952

    accuracy                           0.65     45017
   macro avg       0.67      0.70      0.64     45017
weighted avg       0.75      0.65      0.66     45017
```

After we have a best model we try evaluation with confusion matrix and classification report. Why we choose a recall? First, let's take a deeper look at this confusion matrix.

'The general idea is to count the number of times instances of class A are classified as class B'

We have the result of confusion matrix with Random Forest Classifier algorithm.

- Precision score ==> precision = (TP) / (TP+FP)

Precision is the accuracy model predict a positive or negative class

- Recall Score ==> recall = (TP) / (TP+FN)

Recall is the accuracy of positive or negative predictions from the actual data.

- Accuracy Score

Accuracy is the accuracy of the overall prediction

As we know **zero label for (Property Damage)** and **one label for (Injury Collisions)**. So, we try to optimize class one to optimize the model predict the injury collisions and after that we evaluation the best accuracy score the model. The model would be more preventive action if the actual data is property damage and the model predict the injury collisions. The prediction would be more effective when the model labeling one (injury collisions) for gain awareness from the people who using vehicle in the traffic.

Finally, we have the best model for our case. But, we can more evaluation this model with f1-score and ROC Curve, how the performance our model when we evaluation with that model, is the best model or we have another model to give more best and effective model.