IBM – Coursera

Data Science Specialization

Capstone project - Final report

Road Accident Severity Prediction

Olajide ILELABOYE – 2020

Table of content:

I. Introduction:

This report is for the final course of the Data Science Specialization. A 9-courses series created by IBM, hosted on Coursera platform. The problem and the analysis approach are left for the learner to decide, hence the knowledge of the course is applied to “Road Accident Severity Prediction”

Most urban cities are plague with traffic issue as a result of road accidents across major highways and busy roads. Road accidents sometimes results into fatalities, injuries or just property damage. External factors like weather, road and visibility conditions sometimes reveal clues about severity of such accidents.

The main goal is to provide early-warnings by exploring traffic data provided from the course platform by creating a machine learning model.

The target audience for this report are:

- Potential road users: Motorists, bikers, Cyclists and pedestrians.
- Traffic Controllers and Road administrators.
- Instructors and peers who will grade this project. Or anyone who care to use for learning or research purposes.

II. Data description:

From available data set provided by Coursera, **'*Data-Collision.csv***' from ***GISWEB*** , I will be using attributes like '*WEATHER*' , '*ROADCON*' and '*LIGHTCON*' to determine severity of road accidents. Hence, my target variable will be '*SEVERITYCODE*' These data will be used to train a model and test accuracy of each of the result to determine best suitability to predict future warnings.

The csv metadata file contained useful information about description of each data column;

LIGHTCON: The light conditions during the collision (visibility)

ROADCON: The condition of the road during the collision

SEVERITYCODE: The code that corresponds to the severity of the collision:

3-fatality

2b-serious injury

2-Injury

1-Property damage

0-unknown

WEATHER: A description of the weather conditions during the time of the collision.

The process of collecting and clean data:

- Import data into panda data frame by reading the CSV file.
- Check columns for missing data. Dropped columns not used for analysis.
- Covert data typed of target columns from “object” to numeric forms via encoding.

	SEVERITYCODE	X	Y	OBJECTID	INCKEY	COLDETKEY	REPORTNO	STATUS	ADDRTYPE	INTKEY	...
0	2	-122.323148	47.703140	1	1307	1307	3502005	Matched	Intersection	37475.0	...
1	1	-122.347294	47.647172	2	52200	52200	2607959	Matched	Block	NaN	...
2	1	-122.334540	47.607871	3	26700	26700	1482393	Matched	Block	NaN	...
3	1	-122.334803	47.604803	4	1144	1144	3503937	Matched	Block	NaN	...
4	2	-122.306426	47.545739	5	17700	17700	1807429	Matched	Intersection	34387.0	...

5 rows × 38 columns

Figure 1 - Initial dataset

After clean-up;

```
softdata.head()
```

	SEVERITYCODE	WEATHER	ROADCOND	LIGHTCOND	WEATHER_CAT	ROADCOND_CAT	LIGHTCOND_CAT
0	2	Overcast	Wet	Daylight	4	8	5
1	1	Raining	Wet	Dark - Street Lights On	6	8	2
2	1	Overcast	Dry	Daylight	4	0	5
3	1	Clear	Dry	Daylight	1	0	5
4	2	Raining	Wet	Daylight	6	8	5

Figure 2 - Final dataset

I used python to check clean-up data to see contribution of each factor to severity of accident based on available data.

Results of the data are shown below;

```
df['WEATHER'].value_counts()
```

```
Clear                       111135
Raining                      33145
Overcast                     27714
Unknown                      15091
Snowing                        907
Other                          832
Fog/Smog/Smoke                 569
Sleet/Hail/Freezing Rain       113
Blowing Sand/Dirt               56
Severe Crosswind                25
Partly Cloudy                    5
Name: WEATHER, dtype: int64
```

Figure 3: Weather Conditions

```
df['ROADCOND'].value_counts()
```

```
Dry                124510
Wet                 47474
Unknown             15078
Ice                  1209
Snow/Slush           1004
Other                 132
Standing Water        115
Sand/Mud/Dirt          75
Oil                    64
Name: ROADCOND, dtype: int64
```

Figure 4: Road conditions

```
df['LIGHTCOND'].value_counts()
```

```
Daylight                    116137
Dark - Street Lights On      48507
Unknown                      13473
Dusk                          5902
Dawn                          2502
Dark - No Street Lights       1537
Dark - Street Lights Off      1199
Other                          235
Dark - Unknown Lighting         11
Name: LIGHTCOND, dtype: int64
```

Figure 5 Light Condition

III. Methodology:

There are two categories in supervised machine learning: linear regression and classification. But as the target variable (severity of the car accident) is not continuous, only classification model is available predict the severity namely;

a. K-Nearest Neighbor: The techniques used to predict severity of outcome by finding the most similar data point within a distance (k).

b. Decision Tree: outline layout of possible outcomes from different weather condition.

c. Logistic Regression: Since dataset provides two possible severities, hence, it is good for prediction within two possible outcomes.

The final dataset contained 194,673 observations split into train set (70%) and test set (30%). The train dataset is used to train the model, and the test set is used to test the accuracy using Jaccard, F1-score and Log Loss.

Completed Python code for data science tools used to analyze data can be found here:

https://github.com/isofttouch/Coursera_Capstone/blob/master/Capstone%20Road_Accident_Code_Sheet.ipynb

.

IV. Results:

The following table shows the frequency table of the true value of severity and the forecasted severity using different classification algorithms:

	y_test	KNN_yhat	DT_yhat	LR_yhat
1	41083	58392	58402.0	58402.0
2	17319	10	NaN	NaN

Table 6. Frequency Table of Forecast Results

As seen above, severity of some of the car accident are most likely to cause “Property Damage” frequent in severity code category (1)

Table 7 below shows model evaluation results. K-nearest neighbor and logistic regression have the same Jaccard index and F1-score. Which perform better that Decision tree that only share same Jaccard score but deviated in F1-score.

But KNN perform best compare to others.

Algorithm	Jaccard	F1-score	LogLoss
KNN	0.703	0.581	NA
Decision Tree	0.703	0.413	NA
Logistic Regression	0.703	0.581	0.601

Table 7. Evaluation Result

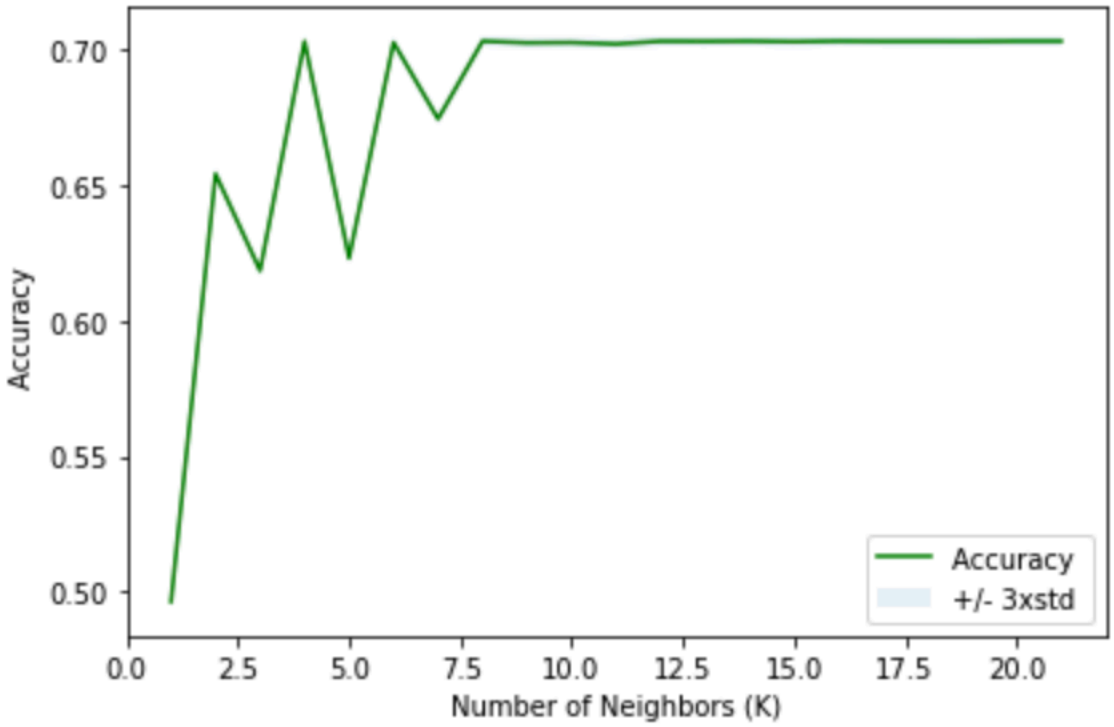

Figure 8 - Accuracy of K-value selection

V. Discussion:

The real challenge is constructing the dataset:

- There are missing information on the dataset. These columns with missing information are dropped during cleaning.

- Characteristic column like ‘SPEEDING’ values would have add more weight to the predictions of severities, it may be necessary to compare data from multiple sources within the same geographical zones.

- In its original form, datasets required are of type “object”, whereas type “integer” or numeric values are necessary for our analysis. Therefore, data types need to be changed or encoded to numerical values useful for appropriate machine learning.

On the other hand, as the target variable (severity of the car accident) is not continuous, classification model (K-nearest neighbor, decision tree and Logistic regression) are used in predicting severity.

VI. Conclusion:

From historical data of weather and road conditions in relation to categorized-severity, it could be concluded that particular weather and road conditions may impact decision about travelling or on road usage which may result in property damage (1) or injury (2) as shown from traffic data.

Link to Presentation on GitHub:

https://github.com/isofttouch/Coursera_Capstone/blob/master/Capstone%20Presentation_Road%20Accident_Severity.pdf

References

1. *GISWEB*
2. Coursera - IBM Data Science (Machine Learning Module)