Seattle Collision Analysis

Zhaojie He

Introduction

This data set describes traffic accidents. Because of the different severity of traffic accidents, ambulances are sometimes needed to help the injured. However, due to the severity of traffic accidents, the traffic police generally need to be judged after arriving at the scene, and it may be too late to decide whether to send an ambulance to the scene after the traffic police finishes the judgment. So a model can be designed to determine in advance whether it is a serious accident based on the weather at the location of the accident, the blockage after the accident and other factors. If it is determined to be a serious accident, an ambulance will be sent to the scene at the same time as the traffic police to rescue the injured as soon as possible.

This is a dataset of car accidents from 2004 to the present in Seattle. This data set includes 38 attributes such as the severity of the accident, the number of casualties, the weather conditions at the time of the accident, and the latitude and longitude of the accident location. I will use decision tree classifier to predict the severity code of this dataset.

Analysis

The dataset has 38 attributes. But not all of them can be used as a factor in the decision tree. Thus, as shown as figure 1 below, I delete some attributes.

(1)delete the attributes that are not related to the target

```
In [8]: RITYDESC','EXCEPTRSNDESC','INCDATE','INCDTTM','SDOT_COLDESC','ST_COLDESC','SEGLANEKEY','CROSSWALKKEY','SEVERITYCODE.1','EXCEPTRSNCODE','SPEEDING','PEDROWNOTGRNT','INATTENTIONIND','LOCATION','SDOTCOLNUM','ST_COLCODE','REPORTNO'

In [9]: df_d.head()
```

Out[9]:

	SEVERITYCODE	STATUS	ADDRTYPE	COLLISIONTYPE	PERSONCOUNT	PEDCOUNT	PEDCYLCOUNT	VEHCOUNT	JUNCTIONTYPE	SDOT_COLCODE	UNDERINFL	WEATHER	ROADCOND	LIGHTCOND	HITPARKEDCAR
0	2	Matched	Intersection	Angles	2	0	0	2	At Intersection (intersection related)	11	N	Overcast	Wet	Daylight	N
1	1	Matched	Block	Sideswipe	2	0	0	2	Mid-Block (not related to intersection)	16	0	Raining	Wet	Dark - Street Lights On	N
2	1	Matched	Block	Parked Car	4	0	0	3	Mid-Block (not related to intersection)	14	0	Overcast	Dry	Daylight	N
3	1	Matched	Block	Other	3	0	0	3	Mid-Block (not related to intersection)	11	N	Clear	Dry	Daylight	N
4	2	Matched	Intersection	Angles	2	0	0	2	At Intersection (intersection related)	11	0	Raining	Wet	Daylight	N

Figure 1. Clean the dataset

Then, since Sklearn Decision Trees do not handle categorical variables, the categorical variables should be transformed to numeric variables. Figure 2 and 3 shows the comparison of before the transformation and after the transformation.

```
In [18]: X=df_clean.drop('SEVERITYCODE',axis=1).values

In [19]: X[1:5]

Out[19]: array([['Matched', 'Block', 'Sideswipe', 2, 0, 0, 2,
                 'Mid-Block (not related to intersection)', 16, 0, 'Raining',
                 'Wet', 'Dark - Street Lights On', 0],
                ['Matched', 'Block', 'Parked Car', 4, 0, 0, 3,
                 'Mid-Block (not related to intersection)', 14, 0, 'Overcast',
                 'Dry', 'Daylight', 0],
                ['Matched', 'Block', 'Other', 3, 0, 0, 3,
                 'Mid-Block (not related to intersection)', 11, 0, 'Clear', 'Dry',
                 'Daylight', 0],
                ['Matched', 'Intersection', 'Angles', 2, 0, 0, 2,
                 'At Intersection (intersection related)', 11, 0, 'Raining',
                 'Wet', 'Daylight', 0]], dtype=object)
```

Figure 2. Before the transformation

```
In [44]: X[1:5]

Out[44]: array([[0, 1, 9, 2, 0, 0, 2, 4, 16, 0, 6, 8, 2, 0],
                [0, 1, 5, 4, 0, 0, 3, 4, 14, 0, 4, 0, 5, 0],
                [0, 1, 4, 3, 0, 0, 3, 4, 11, 0, 1, 0, 5, 0],
                [0, 2, 0, 2, 0, 0, 2, 1, 11, 0, 6, 8, 5, 0]], dtype=object)
```

Figure 3. After the transformation

Then, divided the dataset into training set and testing set. As figure 4 shown below.

SETTING UP THE DECSION TREE

```
In [36]: from sklearn.model_selection import train_test_split

In [37]: X_trainset, X_testset, y_trainset, y_testset = train_test_split(X, y, test_size=0.3, random_state=3)
```

Figure 4. The split of the dataset

After the split, it is time to build the model with the training set and predict model with the test set. Then calculate the accuracy of the model.

Figure 5. Decision Tree model

From the Figure 5, the accuracy of the model is 0.74.

And finally, plot the tree into visualization.

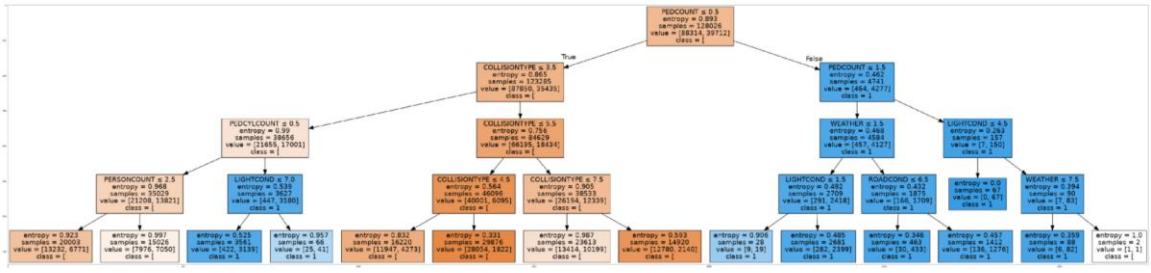

Figure 6. Decision Tree visualization

Conclusion

In conclusion, the decision tree model is qualified, with an accuracy rate of 74.46%. For building a decision tree, there are a total of 13 variables in the input data. But in the end, there are only 6 variables used to build a decision tree. They are PEDCOUNT (The number of pedestrians involved in the collision.), COLLISIONTYPE, PEDCYLCOUNT (The number of bicycles involved in the collision.), LIGHTCOND (The light conditions during the collision.), ROADCOND (The condition of the road during the collision.) and WEATHER. This shows that the road conditions, weather, and light conditions are indeed related to the accident. And through these data, the severity of the accident can be predicted, so that it can be decided in advance whether to dispatch an ambulance.

Of course, this is a decision tree. I think we can try to use Random forest to improve the model. At the same time, you can also try other methods such as KNN and Logistic Regression.