Car accident severity prediction for insurance companies

Joaquín Jiménez-Sauma

Introduction / Business Problem

Predicting the cost, and hence the severity, of claims in an insurance company is a real-life problem that needs to be solved in a more accurate and automated way. Although it is generally considered a smart decision for the car owners to hire a car accident attorney following a crash, in minor collisions, owners can save on lawyer fees by handling the insurance claim themselves. This task makes the insurance companies engage in ample research beforehand by consulting resources and requesting a consultation on how to predict the severity of an accident and in turn the damage that needs to be covered. The overall paper-based process to calculate the severity claim is a tedious task to be completed. This is why insurance companies are continually seeking fresh ideas to improve their claims service for their clients in an automated way. Therefore, predictive analytics is a viable solution to predicting the cost, and hence severity, of claims on the available and historical data.
In this research, we would like to predict the severity of a car accident based on non-human and external factors involving the accident such as Temperature, Wind Chill, Humidity, Pressure, Visibility, Wind Speed and Precipitation, in general, weather conditions.

We want to predict the severity code. This code is interpreted as a number between 1 and 4, where 1 indicates the least impact on traffic.

Background

Insurance companies use a variety of methods to calculate the value of the claim, many of which are different forms of the multiplier method. Unlike lawyers, however, insurance companies rarely use whole numbers as multipliers and instead utilize complex computer algorithms to determine the multiplier.

Generally, there are two reasons people criticize the multiplier method. One criticism is focused on the argument of arbitrary multipliers, meaning that due to the fact that different attorneys use different multipliers, the results are often inconsistent. For example, one attorney may triple the special damages, while another might apply a multiplier of six, opening up a wide gap of variance between the two estimated sums. Additionally, the multiplier method can produce misleading results. The multiplier method can fail to account for more long-term costs and immaterial damage that will affect someone for the rest of their life.

Data Source

This is a United States car accident dataset, which covers 49 states. The accident data are collected from February 2016 to June 2020, using two APIs that provide streaming traffic incident (or event) data.

These APIs broadcast traffic data captured by a variety of entities, such as the US and state departments of transportation, law enforcement agencies, traffic cameras, and traffic sensors within the road-networks. There are about 3.5 million accident records in this dataset.

Source: https://www.kaggle.com/sobhanmoosavi/us-accidents

Sample of data

```
# first rows of data
# this is how it looks

df.head()
```

	Severity	Temperature(F)	Wind_Chill(F)	Humidity(%)	Pressure(in)	Visibility(mi)	Wind_Speed(mph)	Precipitation(in)
0	3	36.9	NaN	91.0	29.68	10.0	NaN	0.02
1	2	37.9	NaN	100.0	29.65	10.0	NaN	0.00
2	2	36.0	33.3	100.0	29.67	10.0	3.5	NaN
3	3	35.1	31.0	96.0	29.64	9.0	4.6	NaN
4	2	36.0	33.3	89.0	29.65	6.0	3.5	NaN

Analysis of Data

Missing values

We have a total of 3,513,617 rows of data.

```
# shape (rows, columns)
# this is how big it is...

df.shape
```

```
(3513617, 8)
```

From this dataset, the number of missing values is:

```
df[['Severity', 'Temperature(F)', 'Wind_Chill(F)', 'Humidity(%)', 'Pressure(in)', 'Visibility(mi)', 'Wind_Speed(mph)','Precipitation(in)']].isnull().sum()
```

```
Severity                0
Temperature(F)      65732
Wind_Chill(F)     1868249
Humidity(%)         69687
Pressure(in)        55882
Visibility(mi)      75856
Wind_Speed(mph)    454609
Precipitation(in)  2025874
dtype: int64
```

Precipitation and Wind Chill contain 57% and 53% of null values respectively. If we delete the rows containing this data, we will lose a larger part of the dataset. It will be better to drop these attributes.

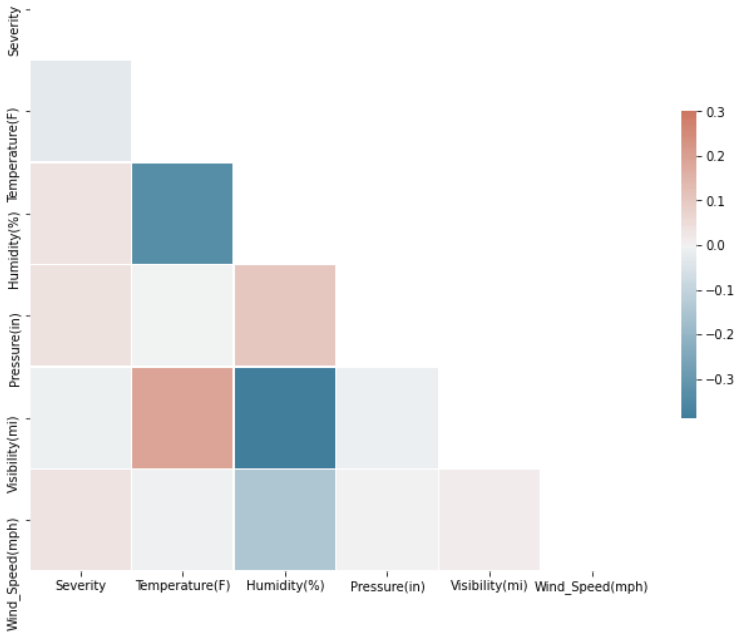
The original dataset contained 3,513,617 rows. After deleting rows with null values we have 3,033,643. We have lost 13.66% of our data. Since this is not a big loss (in my opinion), we can continue using this dataset.

We can use the Pandas method corr() to find the feature other than Severity that is most correlated with Severity.

```
df.corr()['Severity'].sort_values()
```

```
Temperature(F)    -0.028153
Visibility(mi)    -0.006832
Humidity(%)        0.034319
Wind_Speed(mph)    0.035112
Pressure(in)       0.037423
Severity           1.000000
Name: Severity, dtype: float64
```

From a preliminary analysis, we can conclude that Humidity and Temperature are highly related, and Pressure is the feature most directly related to severity. But it is too early to draw definitive conclusions.



Methodology

We are ready to use our dataset and to train our models. We will use the following models:

- K-Nearest Neighbor
- Decision Tree
- Random Forest

An important part of this work is to find the most optimal parameters for each model, so we will find K (for KNN), Max Depth (for Decision Tree) and N Estimators (for Random Forest).

*** Results will be evaluated using Jaccard Score and F1 Similarity.**

As part of the data cleaning process, we decided to reduce the size of the dataset to 5% its size to we can process it in a reasonable amount of time. A random set of rows were retrieved during many iterations and found the results are consistent among iterations.

Results

During our tests we found that the parameters producing the best predictions are:

- KNN: k = 49
- Decision Tree: depth = 4
- Random Forest: estimators = 48

Results showing prediction and evaluation of each model:

Algorithm	Jaccard	F1-score	Precision
KNN	0.678050	0.551332	0.678050
Decision Tree	0.679335	0.550211	0.679335
Random Forest	0.641098	0.586707	0.643636

Conclusions

- After presenting the problem, we found a training dataset and performed an exploratory analysis to understand it.
- We developed models using K-Nearest Neighbor, Decision Tree and Random Forest and found the best training parameters as well.
- Decision Tree model seems to perform better and to have better evaluation than the others.
- Weather conditions can help predict the severity of an accident.