Introduction

In this IBM Data Science Professional Certificate at Coursera, the final Capstone Project consist of predicting the severity of a car crash using a real world dataset from the city of Seattle. There are plenty of business applications for these project from government using that insights collected to know the most dangerous places in the city to insurance companies that could use this to offer more customizable packages to their clients.

Data Understanding

At Coursera we were given a dataset from Seattle Open Data and here are the attributes that were included in the dataset, also their data type, length and description. Attributes in **bold** were the ones that got selected as features for the project.

Attribute	Data type and length	Description
SEVERITYCODE	Text, 100	A code that corresponds to the severity of the collision: • 1—property damage • 2—injury
X	Not specified	Not specified
Y	Not specified	Not specified
OBJECTID	Not specified	Not specified
INCKEY	Long	Not specified
COLDETKEY	Long	Not specified
REPORTNO	Not specified	Not specified
STATUS	ObjectID	ESRI unique identifier
ADDRTYPE	Text, 12	Collision address type: • Alley • Block • Intersection
INTKEY	Double	Key that corresponds to the intersection associated with a collision
LOCATION	Text, 255	Description of the general location of the collision
EXCEPTRSNCODE	Text, 10	Not specified
EXCEPTRSNDESC	Text, 300	Not specified
SEVERITYDESC	Text	A detailed description of the severity of the collision
COLLISIONTYPE	Text, 300	Collision type
PERSONCOUNT	Double	The total number of people involved in the collision
PEDCOUNT	Double	The number of pedestrians involved in the collision. This is entered by the state.
PEDCYLCOUNT	Double	The number of bicycles involved in the collision. This is entered by the state.
VEHCOUNT	Double	The number of vehicles involved in the collision. This is entered by the state.
INCDATE	Date	The date of the incident.

INCDTTM	Text, 30	The date and time of the incident.
JUNCTIONTYPE	Text, 300	Category of junction at which collision took place
SDOT_COLCODE	Text, 10	A code given to the collision by SDOT.
SDOT_COLDESC	Text, 300	A description of the collision corresponding to the collision code.
INATTENTIONIND	Text, 1	Whether or not collision was due to inattention. (Y/N)
UNDERINFL	Text, 10	Whether or not a driver involved was under the influence of drugs or alcohol.
WEATHER	Text, 300	A description of the weather conditions during the time of the collision.
ROADCOND	Text, 300	The condition of the road during the collision.
LIGHTCOND	Text, 300	The light conditions during the collision.
PEDROWNOTGRNT	Text, 1	Whether or not the pedestrian right of way was not granted. (Y/N)
SDOTCOLNUM	Text, 10	A code provided by the state that describes the collision.
SPEEDING	Text, 1	Whether or not speeding was a factor in the collision. (Y/N)
ST_COLCODE	Text, 10	A code provided by the state that describes the collision.
ST_COLDESC	Text, 300	A description that corresponds to the state's coding designation.
SEGLANEKEY	Long	A key for the lane segment in which the collision occurred.
CROSSWALKKEY	Long	A key for the crosswalk at which the collision occurred.
HITPARKEDCAR	Text, 1	Whether or not the collision involved hitting a parked car. (Y/N)

Also, we can see in the following table the number of non-null values for every attribute.

```
SEVERITYCODE      194673 non-null int64
X                 189339 non-null float64
Y                 189339 non-null float64
OBJECTID          194673 non-null int64
INCKEY            194673 non-null int64
COLDETKEY         194673 non-null int64
REPORTNO          194673 non-null object
STATUS            194673 non-null object
ADDRTYPE          192747 non-null object
INTKEY            65070 non-null float64
LOCATION          191996 non-null object
EXCEPTRSNCODE     84811 non-null object
EXCEPTRSNDESC     5638 non-null object
SEVERITYCODE.1    194673 non-null int64
SEVERITYDESC      194673 non-null object
COLLISIONTYPE     189769 non-null object
PERSONCOUNT       194673 non-null int64
PEDCOUNT          194673 non-null int64
PEDCYLCOUNT       194673 non-null int64
VEHCOUNT          194673 non-null int64
INCDATE           194673 non-null object
INCDTTM           194673 non-null object
JUNCTIONTYPE      188344 non-null object
SDOT_COLCODE      194673 non-null int64
SDOT_COLDESC      194673 non-null object
INATTENTIONIND    29805 non-null object
UNDERINFL         189789 non-null object
WEATHER           189592 non-null object
ROADCOND          189661 non-null object
LIGHTCOND         189503 non-null object
PEDROWNOTGRNT     4667 non-null object
SDOTCOLNUM        114936 non-null float64
SPEEDING          9333 non-null object
ST_COLCODE        194655 non-null object
ST_COLDESC        189769 non-null object
SEGLANEKEY        194673 non-null int64
CROSSWALKKEY      194673 non-null int64
HITPARKEDCAR      194673 non-null object
```

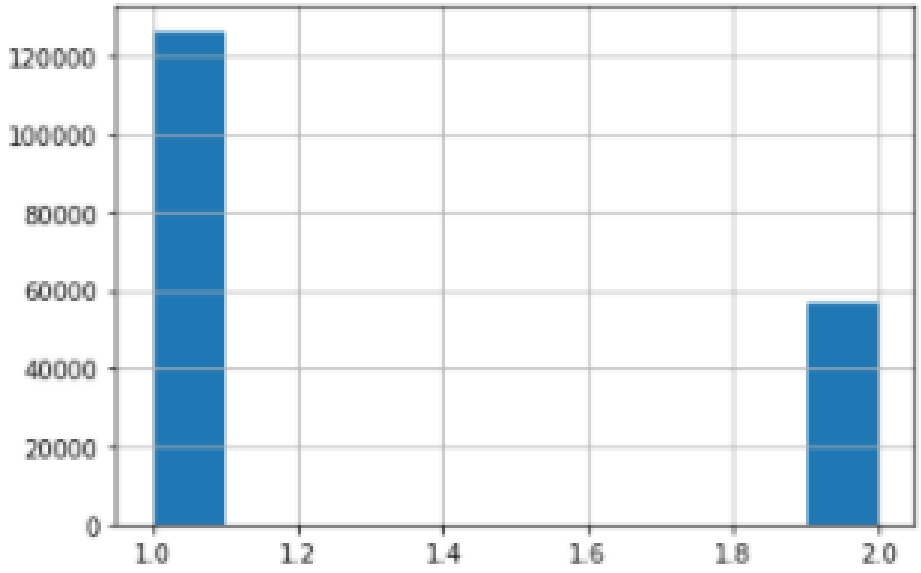
Data Preparation

After features were selected we had to handle missing values. Here is our approach.

Feature	Non-null values	Null values approach
SEVERITYCODE	194673	No null values.
ADDRTYPE	192747	Since there were only 1926 null values, we decided to drop them.
COLLISIONTYPE	189769	Since there were only 4797 null values, we decided to drop them.
PEDCYLCOUNT	194673	No null values
VEHCOUNT	194673	No null values.
INATTENTIONIND	29805	Since we believe inattention would be evident, we made the assumption that null values were negative.
UNDERINFL	189789	We made an assumption that null values weren't under influence of substances.
WEATHER	189592	We made the assumption that null values were on clear days.
ROADCOND	189661	We made the assumption that null values were on clear roads.
LIGHTCOND	189503	We made the assumption that null values were on daylight.
PEDROWNOTGRNT	4667	We made the assumption that null values were cases in which pedestrian permission wasn't granted.
SPEEDING	9333	We made the assumption that null values weren't cases of speeding.

Assumptions made on "WEATHER", "ROADCOND" and "LIGHTCOND" were made because our reasoning was that people preferably decides to drive on the better conditions.

We made a plot in order to know how balanced was our target (“SEVERITYCODE”) and we discovered that it was skewed and that we had an inbalanced dataset.



In order to balance our target outcomes, we decided to use the SMOTE (Synthetic Minority Oversampling Technique) algorithm in order to have the same number and that our models could work with balanced outcomes and give us better performance.

Methodology

Since we want to predict if a car cash results in only a property damage or if it is more severe (injury) then we can say we have a classification problem. Classification models used for this project were the following:

- K Nearest Neighbors.
- Decision Tree.
- Support Vector Machine.
- Logistic Regression.

K Nearest Neighbors

Is a simple algorithm that stores all available data points and classifies new cases bases on similarity measures, such as distance.

Decision Tree

Is a model that separates possible outcomes into nodes and then it goes downstream as it breaks down into more nodes.

Support Vector Machine

Is a model that is able to generalize between 2 different classes if the set of labeled data is provided in the training set to the algorithm,

Logistic Regression

Is a model that uses a sigmoid function to obtain probabilities and therefore it can divide data points to differentiate classes.

Results

Models were trained on 70% of the original dataset. With sklearn library it was possible to get accuracy scores that are determined on the test set to evaluate models. We will make our decision to select the best model with the highest F1 score. In the following table the results are presented.

Model	**Jaccard Index**	**F1 score**	**Log loss**
KNN	0.72	0.71	NA
Decision Tree	0.67	0.68	NA
SVM	0.66	0.68	NA
Logistic Regression	0.66	1	0.55

Conclusion

In this project, we built classification models in order to get accurate results trying to predict whether a car crash could only end in a property damage or an injury. We have shown that we the features selected, severity code of a car crash could be predictable. Of all the four models selected for this project, Logistic Regression was the one with the better F1 score, alongside with that advantage that Logistic Regression has the fastest runtime.

Future directions

Even though Logistic Regression was the model with better score, the F1 score of 1 could possibly be flawed and because of that suggestions are pretty welcomed.

Alongside the prediction of severity of a car crash another prediction that could be worth of working on could be an algorithm that returns the probability of the occurrence of a car crash according to determined features. This algorithm could be sold to car makers for them to implement it in their own cars and alert users when it predicts risk of a car crash.