Business Understanding:
Can we use a machine learning model to predict the severity of a road accident based on certain independent variables? For example: an emergency first-response organisation (hospital or fire department) could be interested in such a model: if the severity increases when certain conditions are met, the necessary precautionary measures can be taken to prepare the organisation for the processing of more severe cases. In the case of a hospital it might be possible to prepare an extra trauma team or to start diverting patients to other hospitals if the operating theater or intensive care unit are at maximum capacity.
In will try to create a predictive model using machine learning with as dependent variable (the variable we want to predict) the severity of an accident, and as predictors (the independent variables) the ambient weather, road and lighting conditions at the time of the accident.

Data understanding:
I used the provides CSV file containing the data on more than 160,000 road accidents. Of importance to our problem is the severity score, the dependent variable. In the provided data, only two integers are used to represent te severity of the accident: "1" for minor severity, and "2" for a more severe accident.
Since we are interested in the physical conditions (road conditions, weather and ambient light) I have excluded all other columns from the table with the independent variables.

Data Preparation:
I started by slicing out the columns that are interesting to our model:

```
dfp= df[['SEVERITYCODE','WEATHER', 'ROADCOND', 'LIGHTCOND']]
```

After removing rows with unknown values and certain other string categories that are not useful for our purpose ("unknown", "other"), we were left with an unbalanced data sheet:

```
dfp['SEVERITYCODE'].value_counts()
```

Out[583]:

```
1    114274
2     55683
```

To make the modelling smoother, I balanced the data for severity index (so we have an equal amount of rows with value 1 and 2). I also reduced the total sample size to 20,000 to speed up the modelling speed.
To be able to use the machine learning models, I then created dummy columns corresponding to the individual values present in the columns `'WEATHER', 'ROADCOND', 'LIGHTCOND':`

We end up with the following columns, whose values "0" or "1" will become the independent variables we will use to predict the accident severity:

```
Index(['Dark - No Street Lights', 'Dark - Street Lights Off',
       'Dark - Street Lights On', 'Dawn', 'Daylight', 'Dusk', 'Dry', 'Ice',
       'Oil', 'Sand/Mud/Dirt', 'Snow/Slush', 'Standing Water', 'Wet',
       'Blowing Sand/Dirt', 'Clear', 'Fog/Smog/Smoke', 'Overcast',
       'Partly Cloudy', 'Raining', 'Severe Crosswind',
       'Sleet/Hail/Freezing Rain', 'Snowing'],
      dtype='object')
```

A training and test set were created to use while modelling (see below).

Modeling:

Our problem is essentially a classification problem with categorical independent and dependent variables. I will use 4 popular types of models: K nearest neighbors, logistic regression, support vector machines and decision trees, using a Jupyter notebook and the free scikit learn (sklearn) machine learning library.

Evaluation:
Each model was run multiple times and the optimal parameters were selected using the F-1 score and Jaccard similarity score as compared to the "test" set:

- KNN: k=3
- SVM: kernel='rbf'
- Logistic regression: solver = 'liblinear'
- Decision trees: max_depth = 8

The resulting "best models" were then compared to the entire data-set (before down-samoling) of 160,000+ rows, and the models were then compared:

Algorithm	Jaccard	F1-score	Logloss
KNN	0.55	0.55	NA
Decision Tree	0.44	0.44	NA
SVM	0.44	0.43	NA
Logistic Regression	0.45	0.45	0.69

Overall, our KNN model seems to be the most valuable.
The F1 score of our chosen model is higher than 0, but is not truely impressive. But then again: "All models are wrong, but some are useful." We will use this model to predict the accident outcome with certain values for x.

Deployment:
When our customer (e.g. Hospital) feeds the chosen model (the KNN model) certain arrays, we get a prediction of the accident severity:
Example: an accident happening in the dark, icy roads and during snow: "2"
Example: an accident during daylight, on dry roads and in clear weather: "1"
So our model predicted a severe outcome in bad weather and road conditions, something we can explain from a 'mechanistic' point of view.

Conclusion:
Using our KNN model we can predict the severity of an accident using weather, road and lighting conditions as parameters. The accuracy of our model is somewhat disappointing, however, so it can only be used as a rough estimate. Nonetheless, in larger urban areas with multiple accidents per day, a rough estimate might be enough to make the decision to divert resources from non acute services to trauma first response when certain physical conditions are met.

Acknowledgment:
I copypasted code for the ML models from a Githubber (https://github.com/Thomas-George-T/IBM-Data-Science-Professional-Certification) because his code is excellent and clearly structured. So thank you, Thomas George.