# Coursera Capstone Project:

# Predicting car accidents severity from data acquired in Seattle from 2004 to present days

# BUSINESS UNDERSTANDING

- Data analysis on a dataset containing reports of accidents in Seattle from 2004 to present days, from the data preparation to the deployment of a machine learning model

- Stakeholder: the city of Seattle, more specifically the Seattle Department of Transportation, the Traffic Management Department, and the Police

- Goal: build a model to predict the severity of an accident giving particular external conditions, such as the type of location, the weather, the road conditions, or the light. The ability to predict an accident could be used to prevent them and save lives.

# DATA UNDERSTANDING

- Dataset downloaded from the official Seattle GeoData website (https://data-seattlecitygis.opendata.arcgis.com/). Updated weekely, it reports all accidents that have been happening in Seattle from 2004 to present days.
- Csv file, consisting of 221738 rows and 40 columns
- Each column, or attribute, describes the accident: address, type of location, whether it involved cars, bikers or pedestrians, the total number of people involved, as well as external conditions such as the weather, the visibility, the road conditions, or the presence of drivers under influence.
- → Some of these will become the independent variables of our analysis
- Severity of the accident: described by 5 values, corresponding to: fatality, serious injury, inkury, or property damage only, plus a category for unknown severity.
- → This will become the dependent variable (y)

# METHODOLOGY

## Data preparation

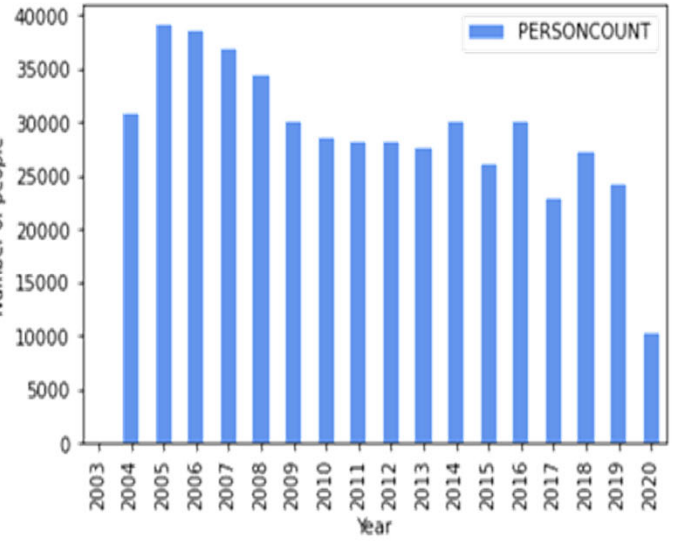
- - We start by reading the dataset into a panda dataframe.
- - This dataframe contains 221738 rows and 40 columns.
- - With the *info()* method, we find out that several columns have missing values.
- - I used *drop()*, *replace()* and *dropna()* method to get rid of columns (attributes) that are not needed for our analysis, as well as missing values.

## Data exploration

- Data visualization by basic plotting: useful to look at possible correlations between the attributes.

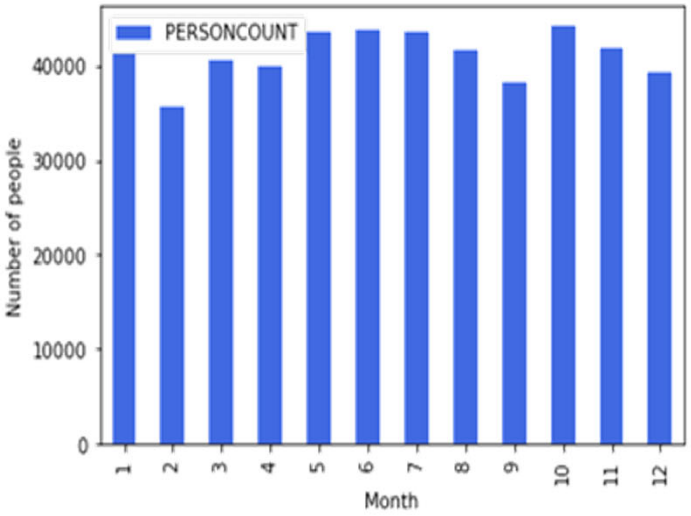
For instance, we can start by looking at the overall number of people involved in accidents during the years, or by month, or by day of the week (0 to 6 meaning Monday to Sunday):

Years: 2004 to now



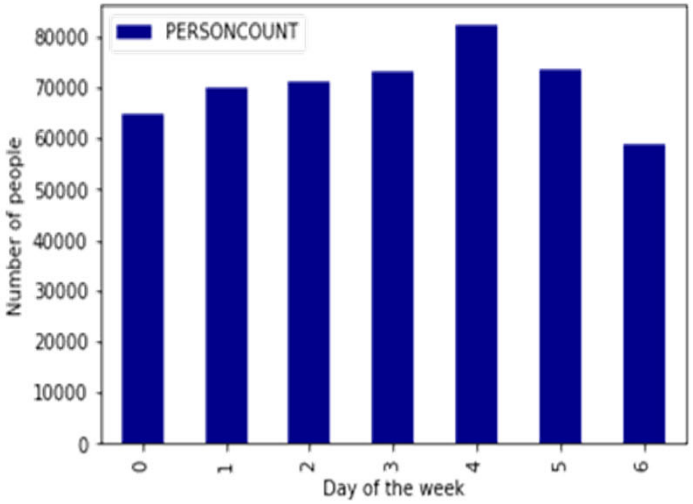

There is a trend: accidents are decreasing (but still happening in 2020 with the lockdown)

Months




There are some differences between months, but not a specific pattern

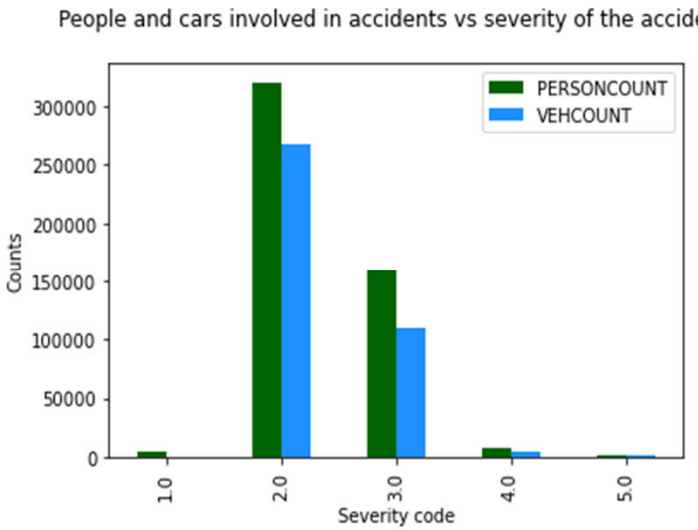
Day of the week

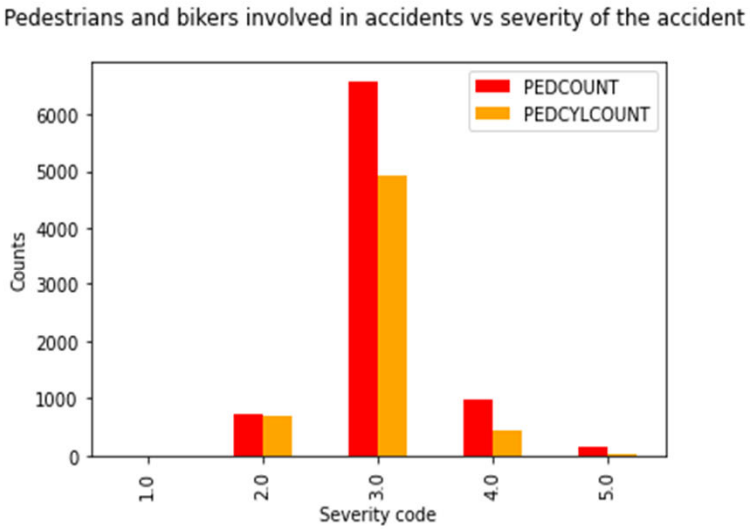



Accidents seem to happen more frequently on Fridays

What about the number of car drivers, bikers and pedestrians as a function of the severity of the accident?

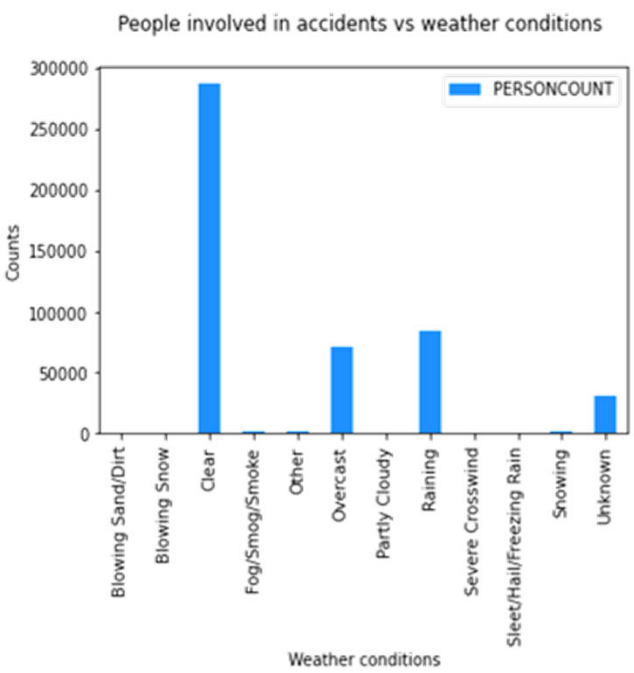
Severity from 1 to 5 meaning: unknown, damaged property, injury, severe injury, fatality

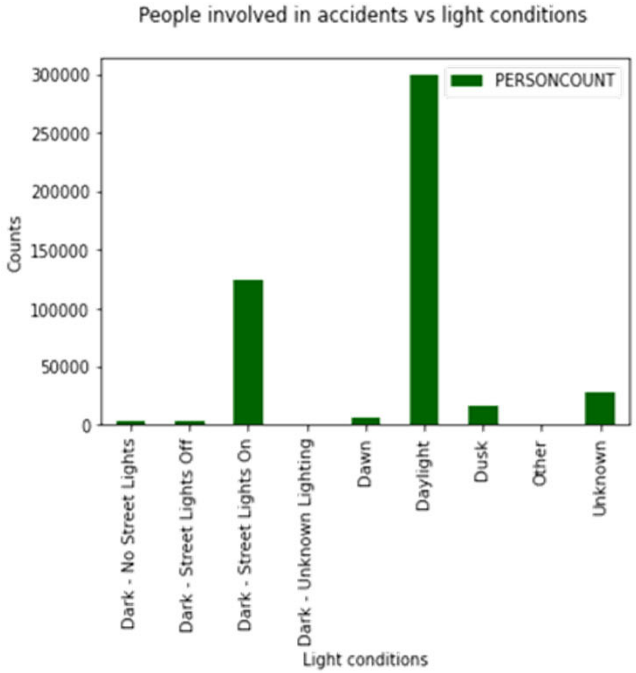



The severity is higher for bikers and pedestrians rather than for car drivers.

Let’s now take a look at how the road, light and weather conditions affect the overall number of people involved in the accident



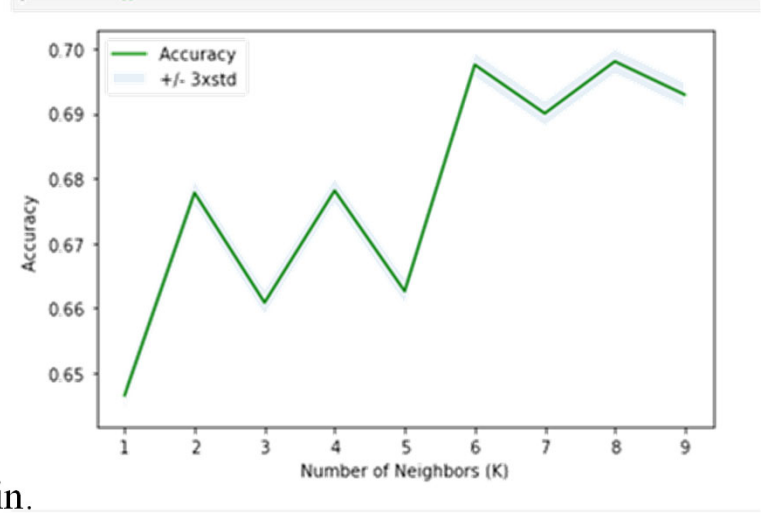


External conditions indeed have an impact on the accident, and the majority of the accident are caused by a limited number of conditions.

## Machine learning models

- I start by defining a new dataset corresponding to the attributes that I want to use in the model as X (independent variables), such as weather, light and road conditions, etc.
- As a dependent variable Y we will use the severity code.
- I perform a train-test splitting of our dataset, keeping the test set size at 30%, and proceed with the normalization of our X dataset.
- Models used: ***decision tree***, ***k-nearest neighbors*** and ***logistic regression***.
- **Evaluation**: performed by two different metrics, the ***Jaccard score*** and the ***F1 score***.
- KNN model: I explore different values of *k*.

The best accuracy is given by *k*=8, which I used to run the model again.



# Results and Discussion

| Algorithm | Jaccard | F1-score |
|---|---|---|
| **Decision Tree** | 0.733372 | 0.684546 |
| **K-Nearest Neighbors** | 0.698035 | 0.684352 |
| **Logistic Regression** | 0.724192 | 0.659623 |

- The three different models have a similar accuracy, between **66%** and **73%**
- → None of them is extremely accurate
- Possible reason: underfitting → not enough data to train the model
- On the other hand, the combination of conditions has indeed an impact on the accidents: it needs improvements but it is a good strategy

# CONCLUSIONS

- Machine learning is a good strategy to improve viability and help preventing accidents
- More data are needed to perform more accurate predictions (with more than 70% probability)
- More accurate reports would be useful to avoid missing values or typos
- The results are useful for the stakeholders, the Seattle Department of Transportation and the Seattle Police:
  - They could train officers to improve the recording of the accidents
  - They could deploy the model to advise drivers and police when the external conditions may lead to an accident.