Predicting of car accidents severity

Report for “Applied Data Science Capstone” course by IBM

Daria Siniatulova

September 30, 2020

CONTENTS

1 Introduction

1.1 Background

According to World Health Organization, every year the lives of approximately 1.35 million people are cut short as a result of a road traffic crash. More than half of all road traffic deaths are among vulnerable road users: pedestrians, cyclists, and motorcyclists. Between 20 and 50 million more people suffer non-fatal injuries, with many incurring a disability as a result of their injury.

Road traffic injuries cause considerable economic losses to individuals, their families, and to nations as a whole. These losses arise from the cost of treatment as well as lost productivity for those killed or disabled by their injuries, and for family members who need to take time off work or school to care for the injured. Road traffic crashes cost most countries 3% of their gross domestic product. Road traffic injuries are the leading cause of death for children and young adults aged 5-29 years. Males are more likely to be involved in road traffic crashes than females. About 73% of all road traffic deaths occur among young males under the age of 25 years who are almost 3 times as likely to be killed in a road traffic crash as young females.

The information on car crashes can be used to understand what factors contribute to different types of accidents, predict their severity category and help City of Seattle prevent them or reduce their damage.

1.2 Problem

The goal of this project is to build several machine learning models to be able to predict severity of consequences for motor vehicles collisions in Seattle using the dataset of previously registered cases and their attributes. This project is based on City of Seattle data on traffic crash, also called a motor vehicle collisions, or car accidents. Motor vehicle collisions occurs when a vehicle collides with another vehicle, pedestrian or cyclist, animal or stationary obstruction.

Project dataset is hosted by the City of Seattle at the open data platform (https://data.seattle.gov), the dataset's attributes description is published at https://www.seattle.gov/Documents/Departments/SDOT/GIS/Collisions_OD.pdf.

Different attributes of the car accidents will be used to build machine learning models and predict severity of accidents.

1.3 Interest

Road traffic injuries can be prevented. Therefore, the possibility to accurately predict whether and how much people risk with their property, health or life while using roads (as car drivers, motorcyclists, passengers, cyclists, or pedestrians) can be used to make important decisions at many levels. For example, to identify most locations of high risk of an accident, to identify main factors of severe car accidents, to adjust construction plans for road infrastructure and bridges, to impose of traffic restrictions in the city center, to plan speed limits at some road parts, etc.

The results of this project could be of interest to governmental regulators and municipal public service officials (for example, in the decision-making process on roads and bridges construction planning), health insurance organizations and medical organizations that work with traffic accidents trauma cases, various businesses (banks, car insurance companies, car dealers, car parts suppliers, repair shops etc.), as well as mass public, media and individual drivers and passengers (for example, on additional precaution measures in particular parts of roads, or types of junctions, or during specific weather).

2 DATA ACQUISITION AND CLEANING

2.1 DATA SOURCES

The dataset is hosted by the City of Seattle at the open data platform. The CSV file about collisions can be obtained via link http://data-seattlecitygis.opendata.arcgis.com/datasets/5b5c745e0f1f48e7a53acec63a0022ab_0.csv

Description of different attributes of the collision cases will be used to build machine learning models and predict severity of accidents.

The file of that description (Collisions_OD.pdf) is available via link https://www.seattle.gov/Documents/Departments/SDOT/GIS/Collisions_OD.pdf.

The dataset includes all types of collisions.

The dataset timeframe: 2004 to Present (September 30, 2020).

Update Cycle: Weekly.

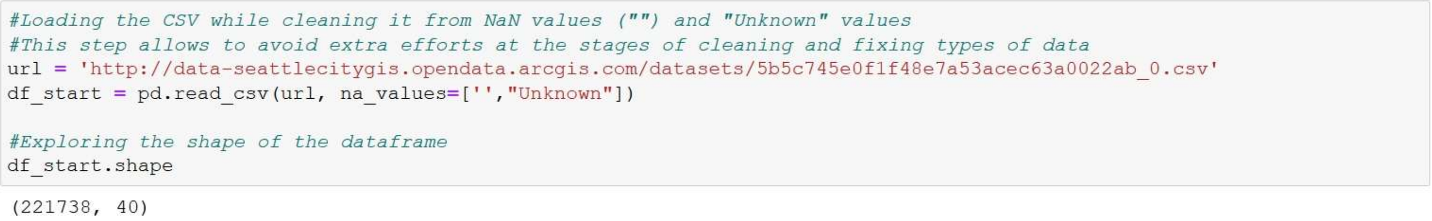
Note: I decided to replace the dataset suggested by course staff that was published at https://s3.us.cloud-object-storage.appdomain.cloud/cf-courses-data/CognitiveClass/DP0701EN/version-2/Data-Collisions.csv to have complete data in terms of dates of accidents (the latest dates available) and detailed severity codes (not only 1 and 2, but also 2b and 3), as I consider this attributes and details to be important for the given assignment.

2.2 DATA CLEANING

At this stage I import libraries, collect the dataset from CSV file and then proceed to determine the attributes (columns) that should be used to train machine learning models and perform cleaning on the important ones.

- Duplicate columns, empty rows, highly similar features and useless report codes were dropped.
- Date and time features were dropped after extracting month and year data.
- Categorical values were replaced with numerical codes.
- Data type mismatch was fixed.

I have loaded the CSV file while cleaning it from NaN values ("") and "Unknown" values.



```
#Loading the CSV while cleaning it from NaN values ("") and "Unknown" values
#This step allows to avoid extra efforts at the stages of cleaning and fixing types of data
url = 'http://data-seattlecitygis.opendata.arcgis.com/datasets/5b5c745e0f1f48e7a53acec63a0022ab_0.csv'
df_start = pd.read_csv(url, na_values=['',"Unknown"])

#Exploring the shape of the dataframe
df_start.shape
```

```
(221738, 40)
```

This step allows me to avoid extra efforts in cleaning and fixing types of data.

The dataset is imbalanced because number of majority cases is 2.2 times bigger than all injuries and fatalities number combined. The "Unknown" severity of accidents (21656 rows) should be dropped at the Data preparation stage.

I decided not to perform under-sampling to reduce the number of majority cases (property damage), if my models would show decent results; and they did, so I skipped this step.

There are a lot of missing values in the dataset, because of lack of record keeping. I used different tactics to deal with this:

- if the number of missing values is not very high, I drop all the data for a particular observations (*dropna* method for *axis=0* and particular *subset*);
- if the variable is important attribute (predictor) for the target variable, and such variable could be fixed using the conditions in another columns, I used masks with different conditions to fill out some missing values (*fillna* method for axis=0);
- if the variable is not a very important attribute (predictor) for the target variable, it could be dropped completely; so when the data goes missing on 60-70 percent of the variable, dropping the variable (*drop* method for *axis=1*) should be considered.

I used *to_datetime* method to load the correct format of collision date (*Timestamp* instead of *string*), and then get the Year and Month from it as separate columns. Accidents grouped by year showed that in the first 7 years number of accidents were gradually decreasing from ~15000 to ~ 12000 per year; average yearly number of collisions was higher than in the following years (2012-2019). In 2020 due to COVID-19 pandemic restrictions, traffic accidents number expected to be significantly lower than in 2019 (statistics of the first half of the year supports this trend). Accidents grouped by month have nor revealed any significant seasonal influence.

After getting rid of missing values, I have corrected type *object* or *float* to type *integer*: for the columns that previously contained them (because NaN is considered as text string) for the following columns:

- ADDRTYPE,
- SEVERITYCODE,
- COLLISIONTYPE,
- JUNCTIONTYPE,
- SDOT_COLCODE,
- UNDERINFL,
- WEATHER,
- ROADCOND,
- LIGHTCOND,
- HITPARKEDCAR.

2.3 Feature selection

As of September 30, 2020 the original dataset contains 221738 rows and 40 columns, and the columns' names are:

'X', 'Y', 'OBJECTID', 'INCKEY', 'COLDETKEY', 'REPORTNO', 'STATUS', 'ADDRTYPE', 'INTKEY', 'LOCATION', 'EXCEPTRSNCODE', 'EXCEPTRSNDESC', 'SEVERITYCODE', 'SEVERITYDESC', 'COLLISIONTYPE', 'PERSONCOUNT', 'PEDCOUNT', 'PEDCYLCOUNT', 'VEHCOUNT', 'INJURIES', 'SERIOUSINJURIES', 'FATALITIES', 'INCDATE', 'INCDTTM', 'JUNCTIONTYPE', 'SDOT_COLCODE', 'SDOT_COLDESC', 'INATTENTIONIND', 'UNDERINFL', 'WEATHER', 'ROADCOND', 'LIGHTCOND', 'PEDROWNOTGRNT', 'SDOTCOLNUM', 'SPEEDING', 'ST_COLCODE', 'ST_COLDESC', 'SEGLANEKEY', 'CROSSWALKKEY', 'HITPARKEDCAR'.

I can group these columns as follows:

- the columns 'X', 'Y', 'ADDRTYPE', 'LOCATION', 'JUNCTIONTYPE', contain attributes on the place of accidents, where 'Y' represents latitude and 'X' represents longitude;
- the columns 'INCDATE' and 'INCDTTM' contain data on time of the accidents;
- the columns 'OBJECTID', 'INCKEY', 'COLDETKEY', 'REPORTNO', 'STATUS', 'INTKEY', 'EXCEPTRSNCODE', 'EXCEPTRSNDESC', 'SDOTCOLNUM', 'SEGLANEKEY' and 'CROSSWALKKEY' contain some non-essential technical data like IDs and codes of reports of the dataset.
- the columns 'SDOT_COLCODE', 'SDOT_COLDESC', 'ST_COLCODE' and 'ST_COLDESC' code and describe the accidents themselves, like what type of participants were involved in the accident, what type of location it was or how the collision between the participants happened;
- the columns 'INATTENTIONIND', 'UNDERINFL', 'PEDROWNOTGRNT', 'SPEEDING' and 'HITPARKEDCAR' contain the data on violations of rules that vehicle's driver was responsible for, like driving under influence of drugs or alcohol, or speeding;
- the columns 'WEATHER', 'ROADCOND', 'LIGHTCOND' describe weather, road and light conditions when the accidents happened, or environmental conditions;
- the columns 'SEVERITYCODE', 'SEVERITYDESC', 'COLLISIONTYPE', 'PERSONCOUNT', 'PEDCOUNT', 'PEDCYLCOUNT', 'VEHCOUNT', 'INJURIES', 'SERIOUSINJURIES', 'FATALITIES' contain valuable data piece on the consequences of these accidents, and the 'SEVERITYCODE' is the labeled data that I need to predict using my machine learning model.

The target value to predict is a severity code of the accident (preferably changed from discrete to continuous values).

Based on definition of our problem and available data, main factors that influence my prediction models would be:

- number of registered accidents;
- environmental conditions;
- drivers’ behavior circumstances;
- participants and specific collision situation.

Upon examining the meaning of each feature, it was clear that there was some redundancy in the features. For example, there was a feature of the severity code (as text strings), and another feature of the severity description (as text strings). These two features contained very similar information (consequences of the accident), with the difference being that the former feature could be easily transformed into numerical value, while the latter feature was simply a comment on that value, duplicating its meaning. In order to fix this, I decided to keep all features that were (or should be encoded to) numerical values that are important for the future machine learning models, and drop redundant ones.

Features (columns) 'INCDATE' and 'INCDTTM' related to the date and time of the accidents were dropped after getting months and years from them.

Features (columns) 'OBJECTID', 'INCKEY', 'COLDETKEY', 'REPORTNO', 'STATUS', 'INTKEY', 'EXCEPTRSNCODE', 'EXCEPTRSNDESC', 'SDOTCOLNUM', 'SEGLANEKEY', 'CROSSWALKKEY', 'SEVERITYDESC' were dropped completely as non-essential.

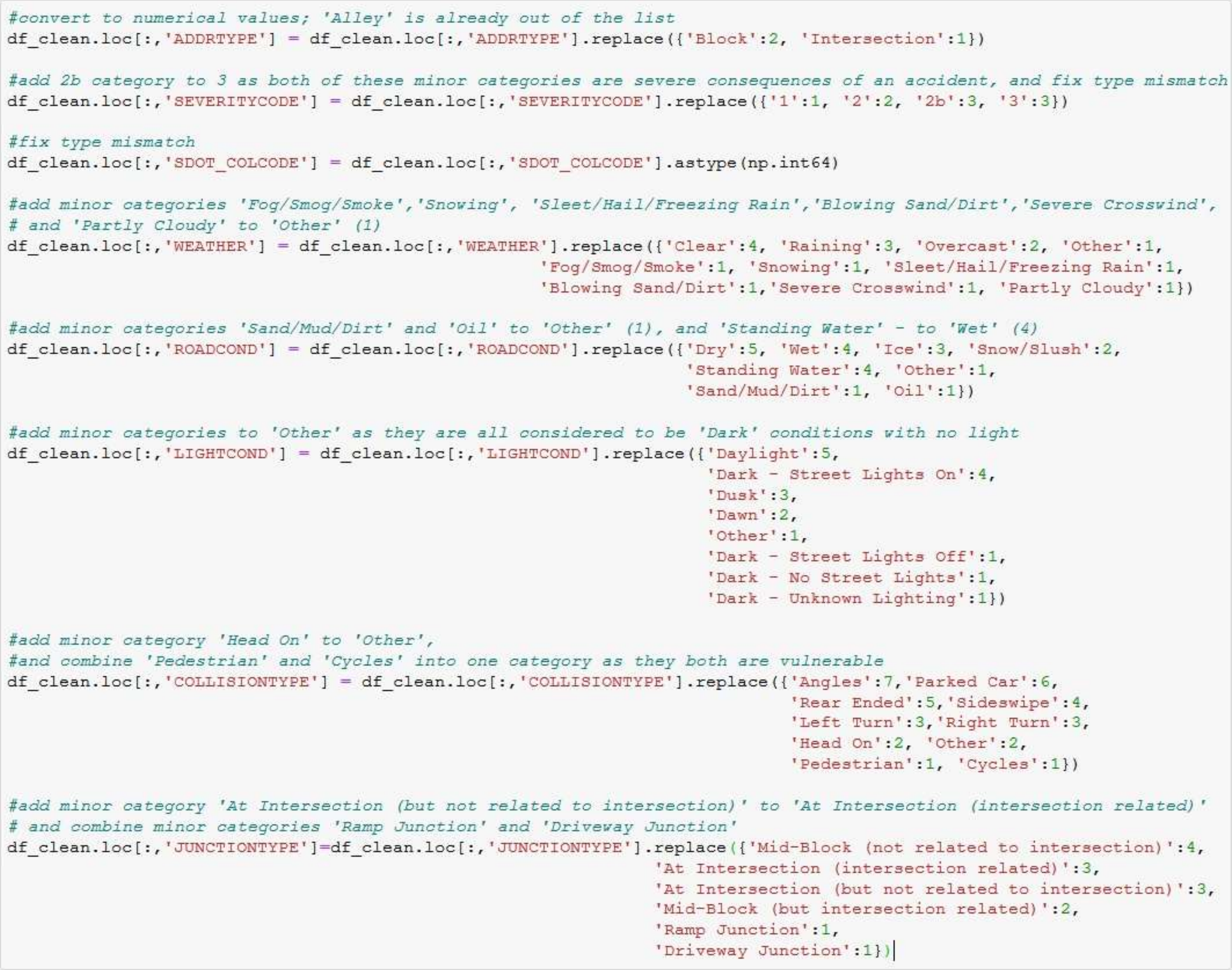
I also created a number of masks that allowed me:

- to filter and drop the rows that lack information on the accidents (SDOT_COLCODE=0), when vehicle was involved (VEHCOUNT>0) but no people were involved (PERSONCOUNT=0), and accidents caused property damage only (45 cases);
- to filter and drop the rows of the accidents when only 1 vehicle was involved, but no people were involved, accidents caused property damage only, and collision type is "Other", so probably they were collisions with stationary obstructions (233 cases):
- to filter and drop the rows of the accidents when no vehicles (VEHCOUNT=0) or people (PERSONCOUNT=0) were involved (4 cases);
- to filter and drop the rows that lack information on the accidents (SDOT_COLCODE=0), and accidents caused property damage only (928 cases);

I used SDOT_COLCODE column values to replace 0 values (errors) in VEHCOUNT and PEDCOUNT columns for particular accidents, when possible:

- if VEHCOUNT=0 and SDOT_COLCODE=24 or 44 (these codes stand collisions when motor vehicle struck pedestrian), VEHCOUNT value should be replaced with 1 (9 cases);
- if PEDCOUNT= 0 and SDOT_COLCODE=24, 44 or 64 (these codes stand collisions when motor vehicle struck pedestrian, driverless vehicle struck pedestrian and cyclist struck pedestrian respectively), then PEDCOUNT should be replaced with 1, at least (66 cases);
- if SDOT_COLCODE is 24, 44 or 64 (these codes stand for motor vehicle struck pedestrian, driverless vehicle struck pedestrian and cyclist struck pedestrian respectively) and PERSONCOUNT is 0, then PERSONCOUNT value should be replaced with 1, at least (179 cases);
- if SDOT_COLCODE is greater than 10 and less than 18 (codes from 11 to 16 stand for the collision accidents when a motor vehicle struck motor vehicle) and VEHCOUNT is 0, VEHCOUNT should be replaced with 2, at least; but the COLLISIONTYPE is identified as "Cycles" as if motor vehicles had nothing to do with it at all. Therefore I just drop those rows as they may contain an error (6 cases);
- now I drop the rest of VEHCOUNT=0 as those rows are not serving the goal of the project - to predict motor vehicles collisions severity code - because there were no motor vehicles involved in these accidents (207 cases).

I have replaced categorical values like address type or weather with numerical codes and fixed data type mismatch:



```
#convert to numerical values; 'Alley' is already out of the list
df_clean.loc[:,'ADDRTYPE'] = df_clean.loc[:,'ADDRTYPE'].replace({'Block':2, 'Intersection':1})

#add 2b category to 3 as both of these minor categories are severe consequences of an accident, and fix type mismatch
df_clean.loc[:,'SEVERITYCODE'] = df_clean.loc[:,'SEVERITYCODE'].replace({'1':1, '2':2, '2b':3, '3':3})

#fix type mismatch
df_clean.loc[:,'SDOT_COLCODE'] = df_clean.loc[:,'SDOT_COLCODE'].astype(np.int64)

#add minor categories 'Fog/Smog/Smoke','Snowing', 'Sleet/Hail/Freezing Rain','Blowing Sand/Dirt','Severe Crosswind',
# and 'Partly Cloudy' to 'Other' (1)
df_clean.loc[:,'WEATHER'] = df_clean.loc[:,'WEATHER'].replace({'Clear':4, 'Raining':3, 'Overcast':2, 'Other':1,
                                                     'Fog/Smog/Smoke':1, 'Snowing':1, 'Sleet/Hail/Freezing Rain':1,
                                                     'Blowing Sand/Dirt':1,'Severe Crosswind':1, 'Partly Cloudy':1})

#add minor categories 'Sand/Mud/Dirt' and 'Oil' to 'Other' (1), and 'Standing Water' - to 'Wet' (4)
df_clean.loc[:,'ROADCOND'] = df_clean.loc[:,'ROADCOND'].replace({'Dry':5, 'Wet':4, 'Ice':3, 'Snow/Slush':2,
                                                                   'Standing Water':4, 'Other':1,
                                                                   'Sand/Mud/Dirt':1, 'Oil':1})

#add minor categories to 'Other' as they are all considered to be 'Dark' conditions with no light
df_clean.loc[:,'LIGHTCOND'] = df_clean.loc[:,'LIGHTCOND'].replace({'Daylight':5,
                                                                     'Dark - Street Lights On':4,
                                                                     'Dusk':3,
                                                                     'Dawn':2,
                                                                     'Other':1,
                                                                     'Dark - Street Lights Off':1,
                                                                     'Dark - No Street Lights':1,
                                                                     'Dark - Unknown Lighting':1})

#add minor category 'Head On' to 'Other',
#and combine 'Pedestrian' and 'Cycles' into one category as they both are vulnerable
df_clean.loc[:,'COLLISIONTYPE'] = df_clean.loc[:,'COLLISIONTYPE'].replace({'Angles':7,'Parked Car':6,
                                                                             'Rear Ended':5,'Sideswipe':4,
                                                                             'Left Turn':3,'Right Turn':3,
                                                                             'Head On':2, 'Other':2,
                                                                             'Pedestrian':1, 'Cycles':1})

#add minor category 'At Intersection (but not related to intersection)' to 'At Intersection (intersection related)'
# and combine minor categories 'Ramp Junction' and 'Driveway Junction'
df_clean.loc[:,'JUNCTIONTYPE']=df_clean.loc[:,'JUNCTIONTYPE'].replace({'Mid-Block (not related to intersection)':4,
                                                                'At Intersection (intersection related)':3,
                                                                'At Intersection (but not related to intersection)':3,
                                                                'Mid-Block (but intersection related)':2,
                                                                'Ramp Junction':1,
                                                                'Driveway Junction':1})
```

After data cleaning, there were 169,089 samples and 22 features in the dataset.

To train the models I used 18 attributes for 135,271 observations of accidents, to test the models I used 18 attributes for 33,818 observations of accidents.

3 METHODOLOGY

The general plan for this project analysis is:

- Load, read and inspect the dataset.
- Fix missing data and type mismatch.
- Investigate relations between accidents severity and attributes.
- Pick relevant attributes and build ML models.
- Assess and compare models' performance.
- Improve models and describe the final model with recommendations.

I plan to use the following main libraries and methods:

- Numpy - to work with arrays.
- Pandas - to work with tabular data.
- Datetime - to extract the data from timestamp.
- Plotly, Matplotlib, Seaborn - to plot the data.
- Scikit-learn - to build and assess ML models.

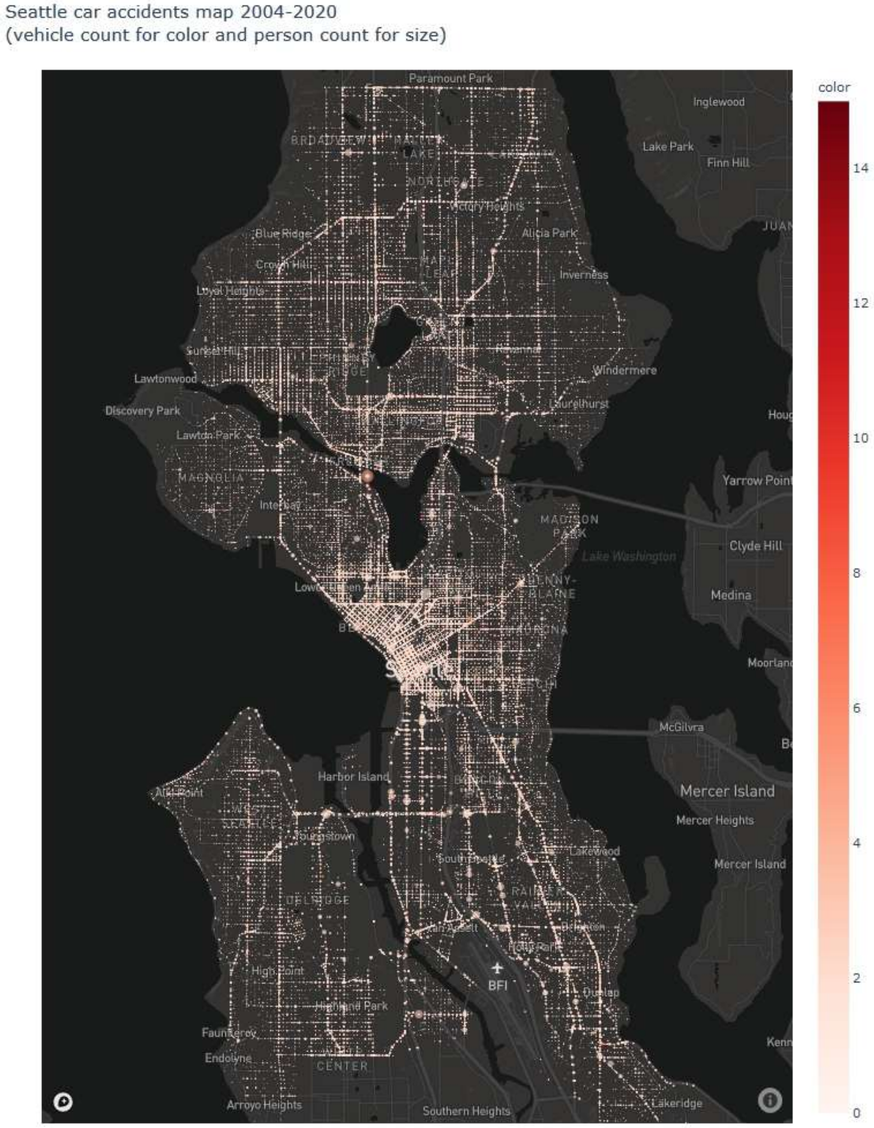
I picked 5 machine learning techniques for the current classification problem:

- Decision Tree.
- Random Forest.
- Logistic Regression.
- Naive Bayes.
- k-Nearest Neighbors.

4 DATA ANALYSIS

According to the business question we need to investigate data, containing information about different types of road accidents, including information about severity and different factors that may have caused the accident. In this project we should pick relevant indicators to build a supervised model that predicts the possible road accident severity with sufficient accuracy.

Exploring the data also implied assessing the condition of chosen attributes by looking for trends, patterns, skewed information, correlations, missing data and errors. Collisions have geospatial coordinates at the intersection or mid-block of a segment. I plot the map of Seattle to check if all of the dots are within the city borders (that is correct).

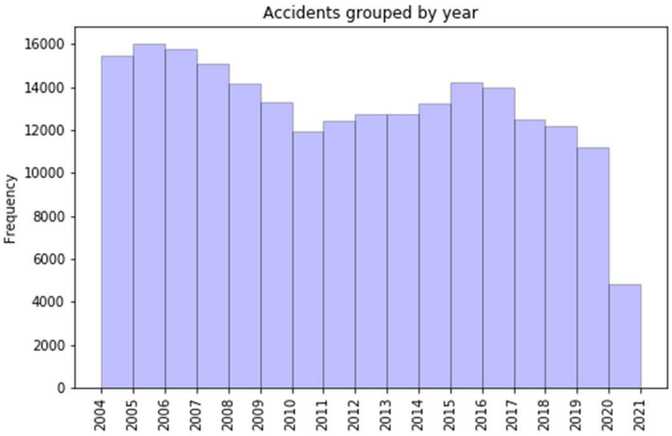


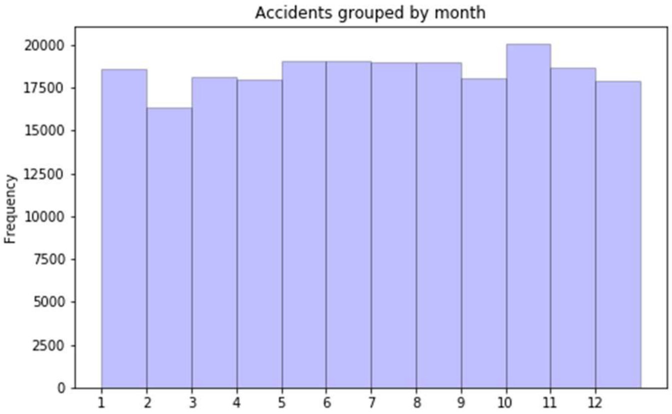
Seattle car accidents map 2004-2020
(vehicle count for color and person count for size)

For example, map shows Aurora bridge crash that happened in September, 2015 when the amphibious vehicle crossed into oncoming traffic, apparently after a mechanical failure, and collide with the college bus. Ninety three people were involved, five people were killed and more than 60 injured. A civil lawsuit filed on behalf of 42 people who were injured or killed in the crash, names the City of Seattle and the State of Washington co-defendants. A median on the bridge could have prevented the fatal crash. This case serves as an example how city planning decisions may prevent severe consequences and save lives.

Accidents grouped by year showed that in the first 7 years (2004 to 2011) number of accidents were gradually decreasing from ~15000 to ~ 12000 per year; average yearly number of collisions was higher than in the following years (2012-2019). In 2015 and 2016 there was a growth in number of accidents, but it was not that significant.

In 2020 due to COVID-19 pandemic restrictions, traffic accidents number expected to be significantly lower than in 2019 (statistics of the first half of the year supports this idea). Accidents grouped by month have nor revealed any significant seasonal influence.

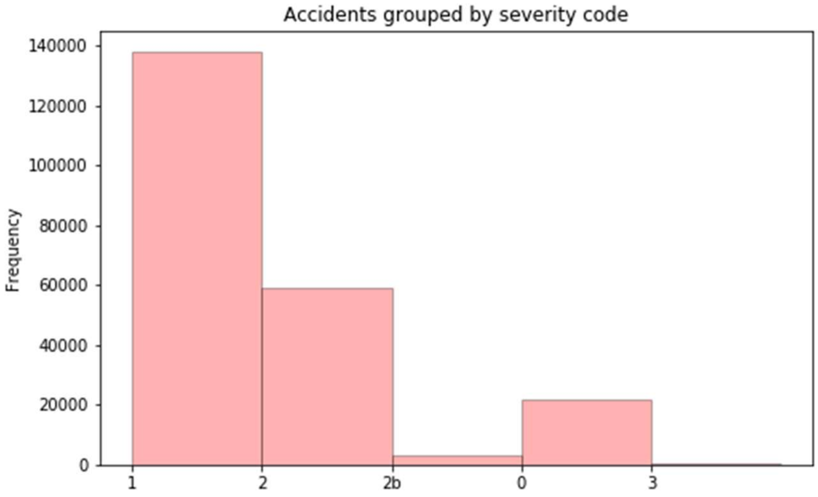




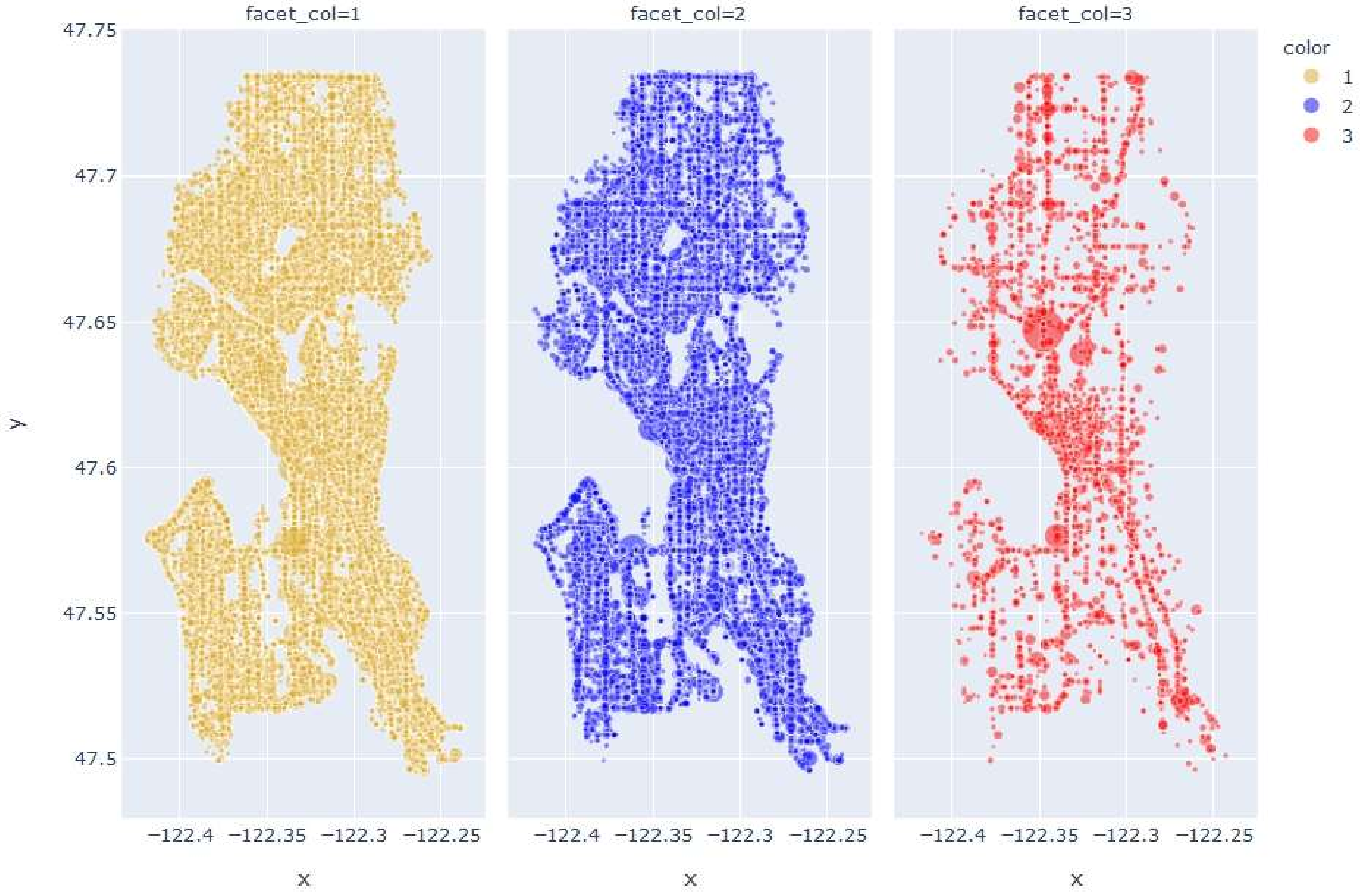
The severity code corresponds to the severity of the collision consequences:

- 3 for fatality (352 cases);
- 2b for serious injury (3,111);
- 2 for injury (58,842);
- 1 for property damage only (137,776);
- 0 for unknown (21,656).

The dataset is imbalanced because number of majority cases (class 1, property damage only) is 2.2 times bigger than all injuries and fatalities cases number combined (classes 2, 2b and 3). I decided not to perform under-sampling to reduce the number of majority cases (property damage), if my models would show decent results. They did, so I skipped this step.

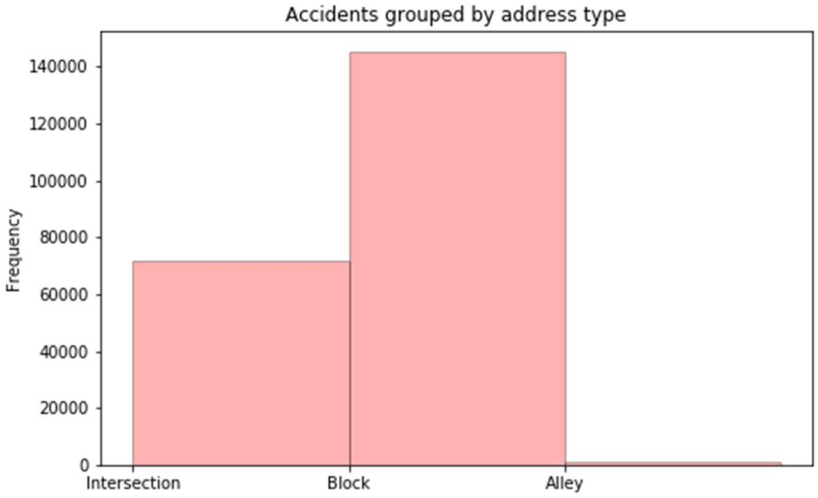


As the map of collisions grouped by severity shows, the severe injuries and fatalities happen all over the city, not in several specific locations (size of the bubble indicates number of people involved).



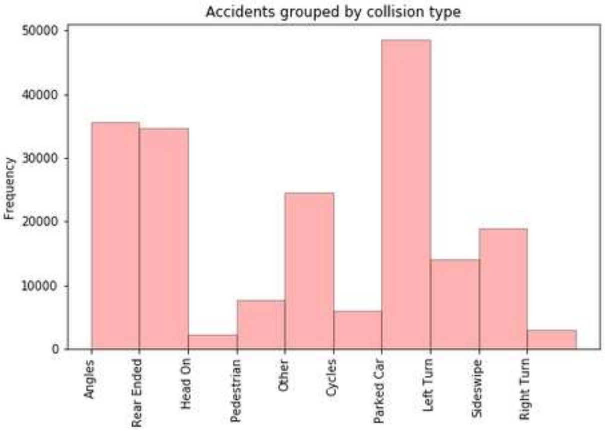
Accidents grouped by address type demonstrate that:

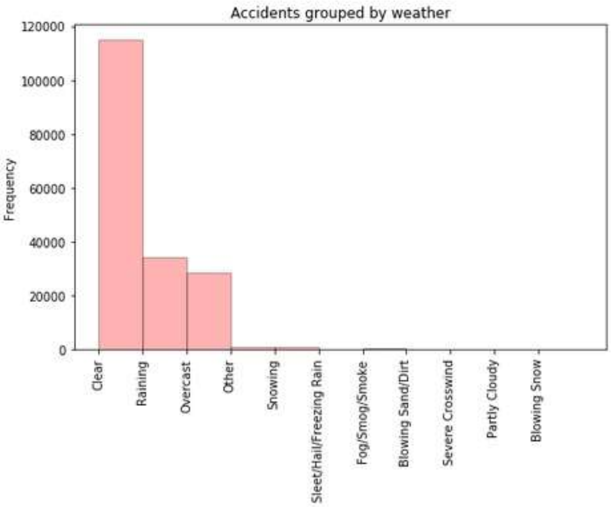
- accidents around the block happen twice more often than accidents at the intersection (145118 vs 72026);
- accidents at the alley (879) are extremely rare (0.4%).



Accidents grouped by collision types demonstrate:

- 80.5% cases are incidents between cars, including:
  - 24.9% involved parked car;
  - 55.6% are collisions between moving cars.
- Only 3.9% involved pedestrians.
- Only 3.0% involved cyclists.



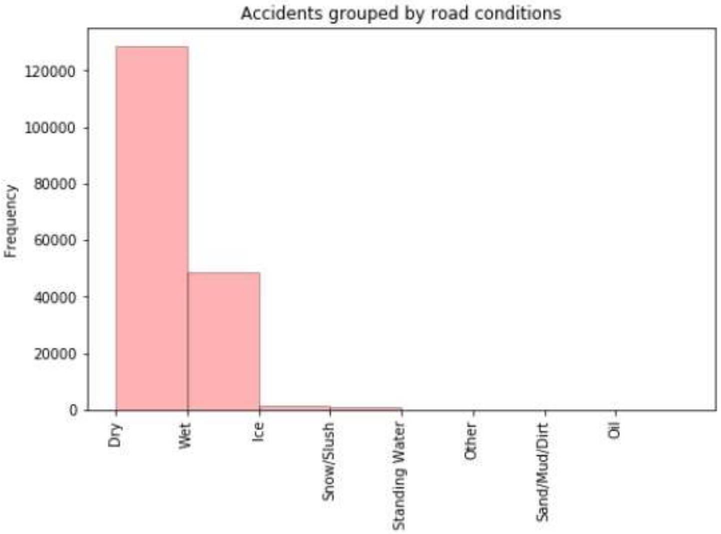


Accidents grouped by weather demonstrate:

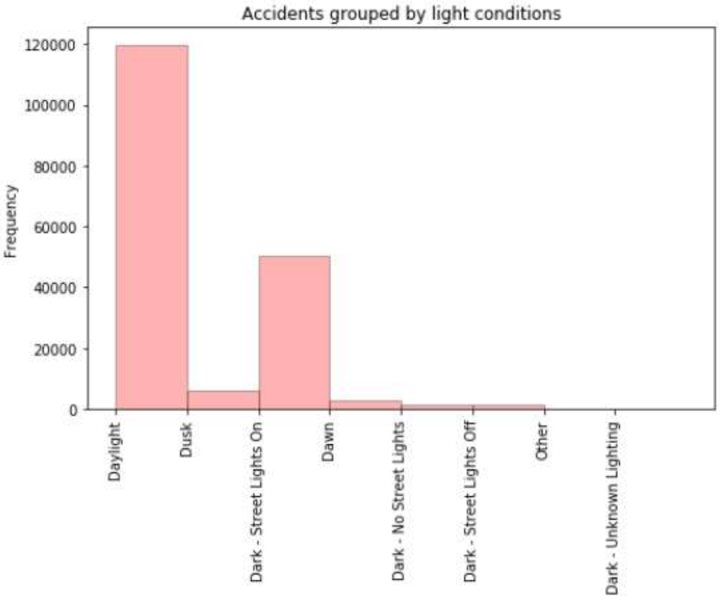
- 64% of traffic accidents happen in clear weather, 19% - in rainy weather and about 16% - in overcast conditions;
- other weather conditions rarely lead to traffic accidents, probably because of both weather events happen not so often in Seattle, or when they do happen the traffic is reduced, or the drivers act more cautious due to difficult weather conditions.

Accidents grouped by road conditions demonstrate:

- 71.4% of accidents happen when the road is dry;
- 27.1% of accidents happen when the road is wet.



Accidents grouped by light conditions show that 66% of collisions happen during the daylight, and collisions during dark time of the day with street lights on are at the second place though are ~2.5 times less frequent.



Accident that happened due to inattention cover only 30,188 cases; due to driving under influence of drugs or alcohol (DUI) – 9,629 cases; due to not granting pedestrian right of way – 5,195 cases; due to speeding – 9,936 cases; and involved hitting of parked car – 12,089 cases.

In conclusion, violations like driver's inattention, speeding, DUI, not granting right of way to a pedestrian or hitting parked car are not necessarily correlated to number or severity of accidents. This could probably be related to limitations of the dataset, as police does not have all the information at the moment of accident registration. Besides, the registration of a collision is possible even without police officers present at the place of accident, as drivers can report them via mobile application. They would not have to wait the police to arrive and get all the details of the accident for the records; for example, in cases “property damage only” they can simply exchange their insurance companies’ contacts and move on.

4.1 IDENTIFICATION OF TARGET VARIABLE (SEVERITY CODE)

The target variable is the severity code, or class of an accident. The original cleaned dataset suggests 4 types of accidents severity (1, 2, 2b and 3), but I decided to unite codes 2b and 3 under class 3, and predict 3 possible accidents outcomes:
1 – for property damage only, 2 – for injuries, 3 – for serious injuries and fatalities.

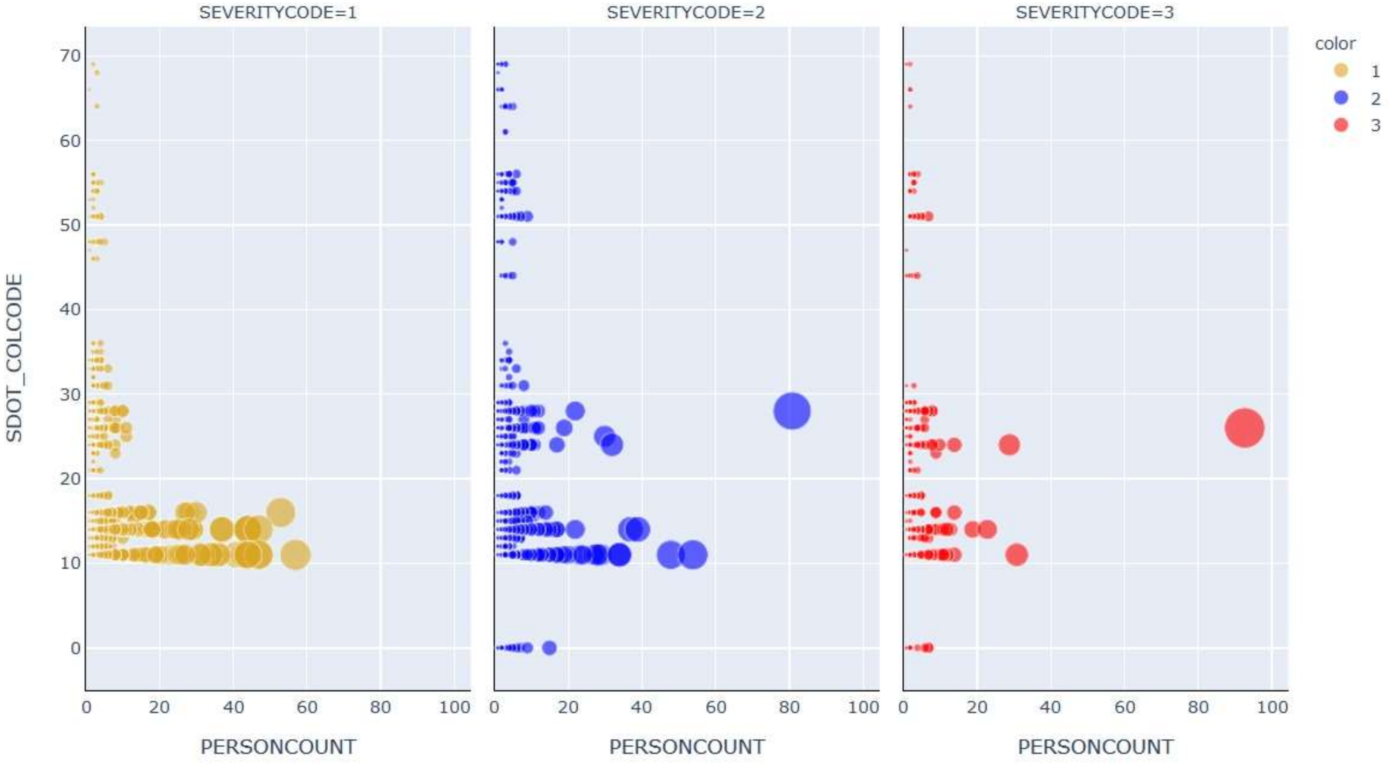
4.2 Relationship between severity, people involved and collision code

I identify the most common road accidents cases in terms of number of people involved in them. I believe this indicator is more important than number of vehicles involved to understand factors that contribute to severe consequences line injuries and fatalities as a result of an accident.

81.7% of accidents (138,212 cases) happened between vehicles when they both were moving from same direction in these situations:

- both were going straight, collision type – sideswipe (80,106 cases, SDOT_COLCODE = 11);
- both were going straight and one stopped, collision type – rear end (48,875 cases, SDOT_COLCODE = 14);
- one was performing right turn, and one was moving straight (8,137 cases, SDOT_COLCODE=16);
- one was performing left turn, and one was moving straight (1,094 cases, SDOT_COLCODE=15).

In these 4 situations 369,267 people were involved in the accidents, or 86.4% of all people in the cleaned dataset.

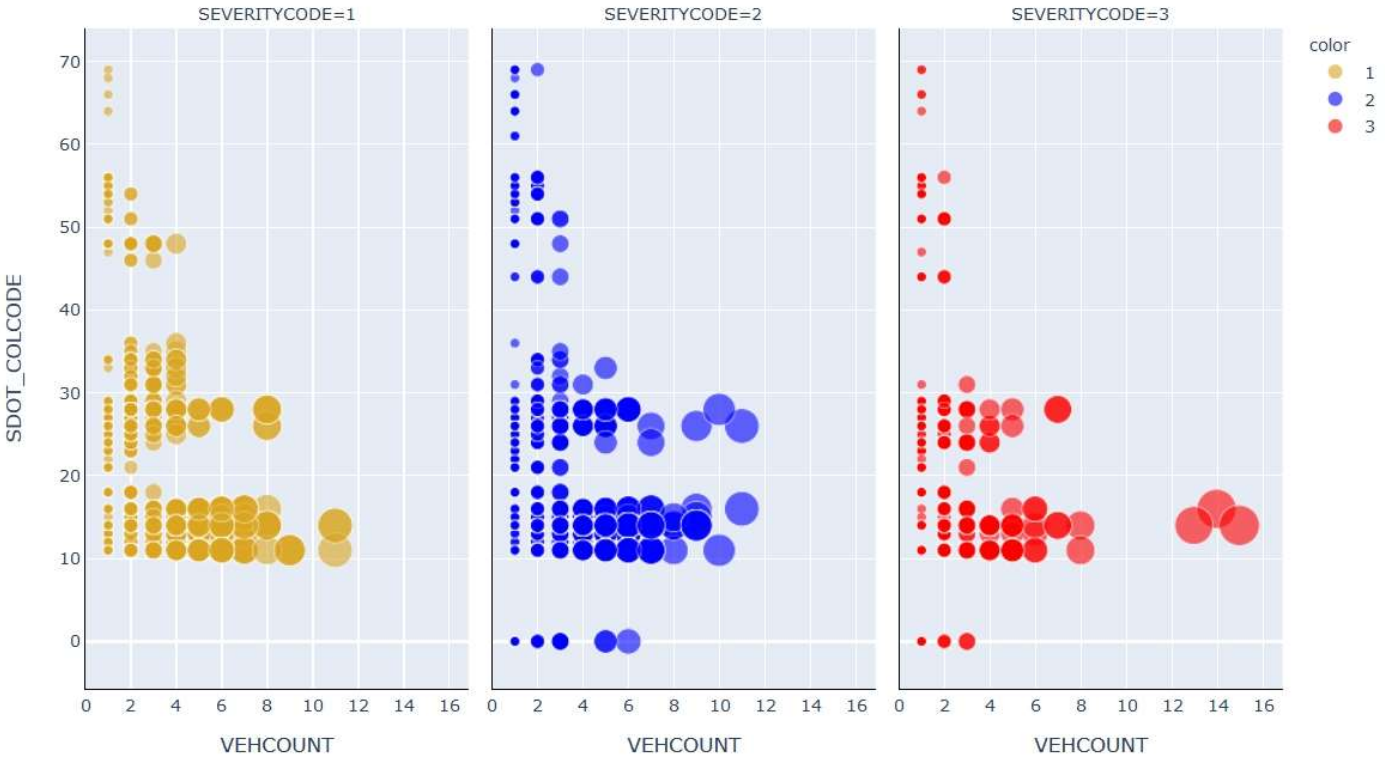


4.3 Relationship between severity, vehicles involved and collision code

Some businesses – like car insurance companies or car repair workshops – may be more interested in number not people but vehicles involved (as those companies are facing the property damage issues as a result of an accident), so same collisions could be described at another angle:

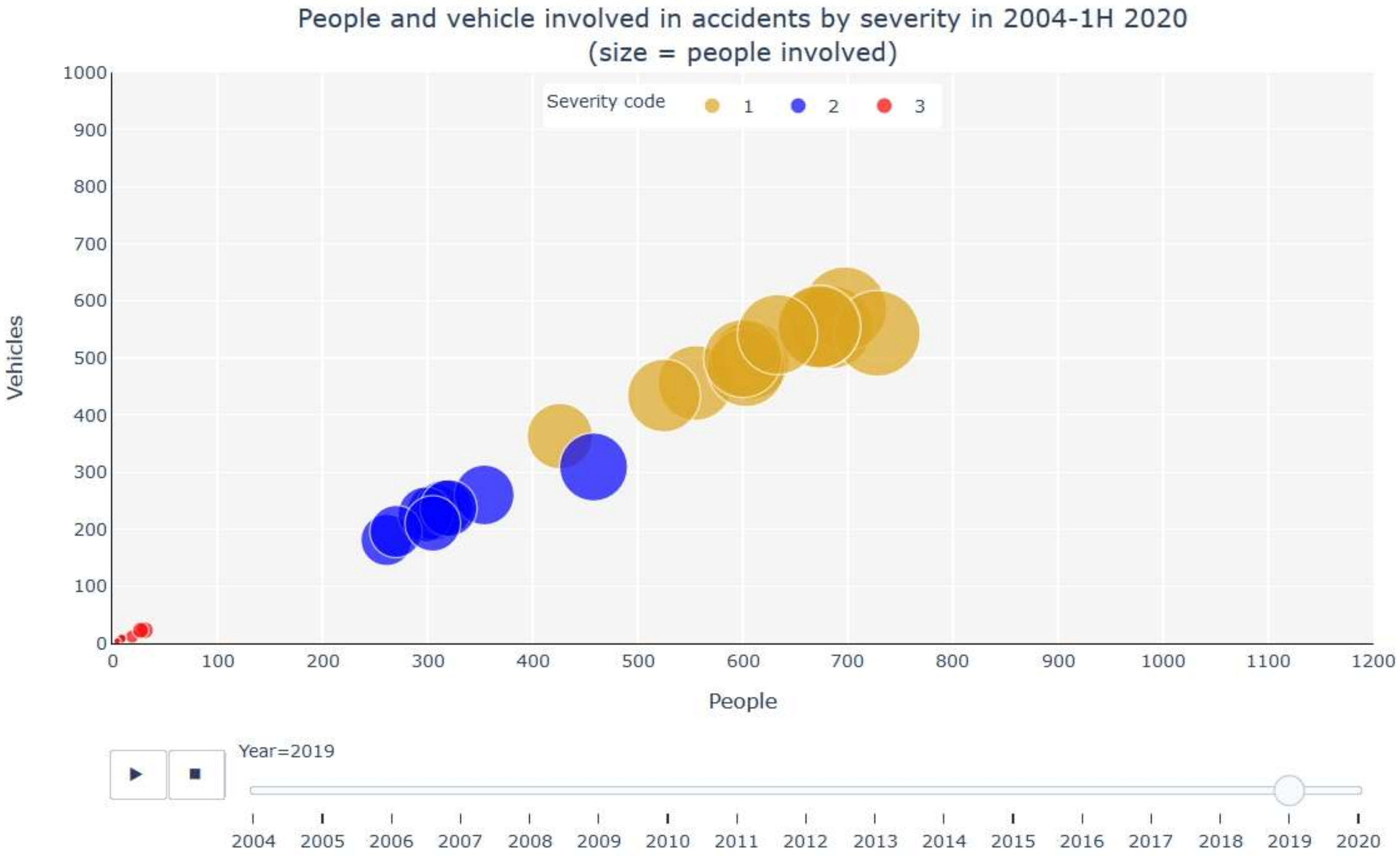
- both were going straight, collision type – sideswipe (166,437 vehicles, SDOT_COLCODE = 11);
- both were going straight and one stopped, collision type – rear end (108,222 vehicles, SDOT_COLCODE = 14);
- one was performing right turn, and one was moving straight (17,439 vehicles, SDOT_COLCODE=16);
- one was performing left turn, and one was moving straight (2,345 vehicles, SDOT_COLCODE=15).

In these 4 situations 294,443 vehicles were involved in the accidents, or 88.3% of all vehicles in the cleaned dataset.



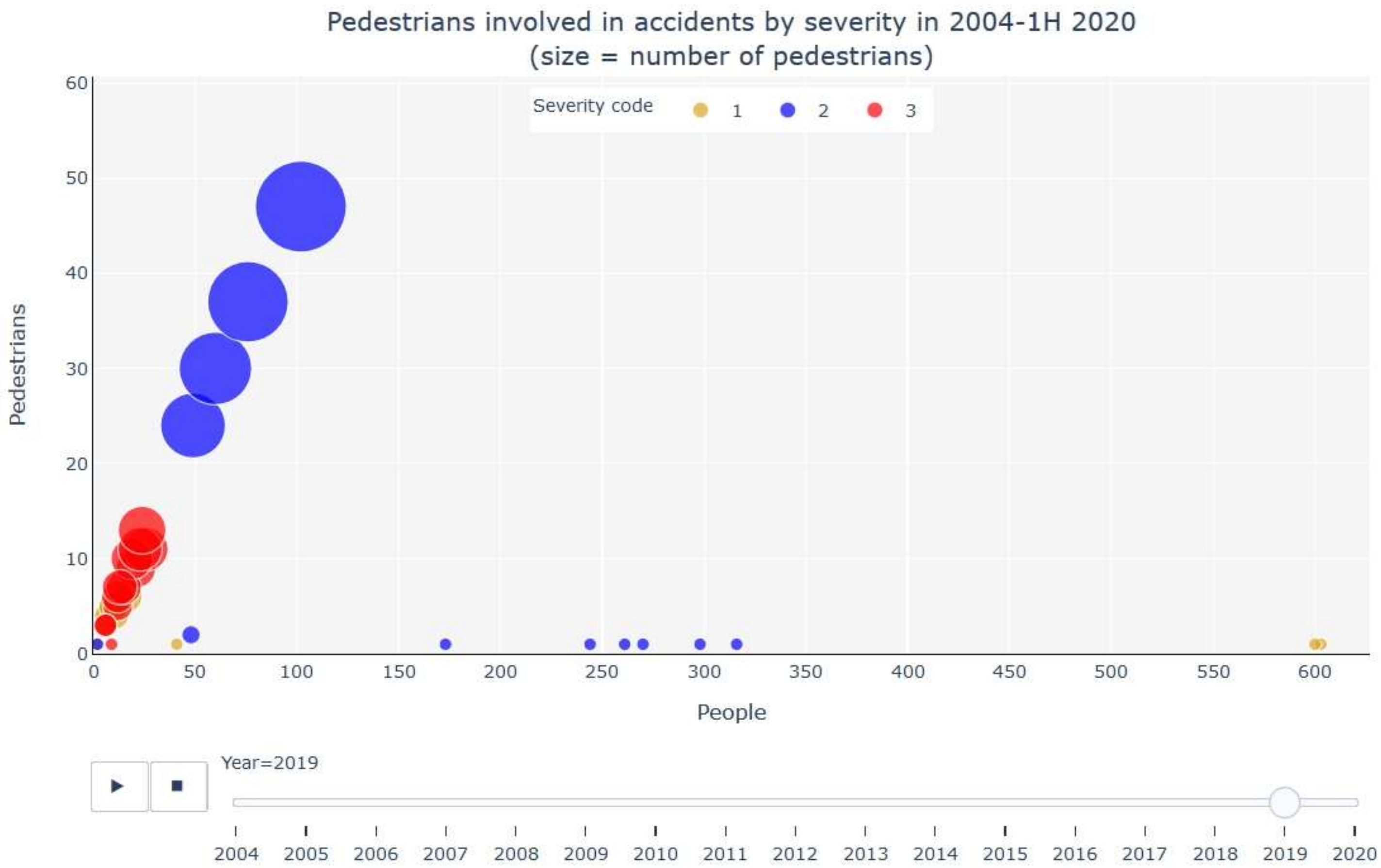
4.4 Relationship between severity, people and vehicles involved

Number of people and vehicles involved in accidents have strong positive correlation. Project notebook contain animated plot to demonstrate dynamic of this relationship since 2004, where every bubble represents accidents of each severity class each month of the year.



4.5 Relationship between severity and pedestrians involved

Pedestrians are the most vulnerable participants of car accidents. If they are involved, the share of cases with injuries, severe injuries and fatalities grows significantly. Project notebook contain animated plot to demonstrate dynamic of this relationship for every month over the years.



5 PREDICTIVE MODELING

The given assignment is a classification problem, as I should predict a type of car accident severity, not a continuous value. Therefore I built and used several classification models, and did not use regression models.

5.1 CLASSIFICATION MODELS

The project "Car accidents severity prediction" is a classification problem. Therefore I applied k-Nearest Neighbors, Random Forest, Logistic Regression, Naive Bayes and Decision Tree techniques to build my machine learning models.

I divided the samples into 3 classes of car accidents severity: 1 – property damage only, 2 – injuries, 3 – severe injuries and fatalities. The number of samples in these classes are unbalanced (class 3 is a minority, class 1 is a majority) so I used weights for each class to balance my models.

I chose Confusion Matrix, Accuracy Score, F1 Score, Precision, Recall, and Support as the metrics.

- Accuracy Score is the most intuitive performance measure and it is simply a ratio of correctly predicted observation to the total observations:

  Accuracy = (True Positives + True Negatives) / (True Positives + False Positives + False Negatives + True Negatives)

- Precision is the ratio of correctly predicted positive observations to the total predicted positive observations, or the ability of the classifier not to label as positive a sample that is negative:

  Precision = True Positives / (True Positives + False Positives)

- Recall (Sensitivity) is the ratio of correctly predicted positive observations to the all observations in actual class, or the ability of the classifier to find all the positive samples:

  Recall = True Positives / (True Positives + False Negatives)

- F1 Score is the weighted average of Precision and Recall; this score takes both false positives and false negatives into account, and is usually more useful than accuracy, especially in case of an uneven class distribution:

  *F1 Score = 2 * (Recall * Precision) / (Recall + Precision)*

5.1.1 Applying standard algorithms and their problems

To train the models I used 18 attributes for 135,271 observations of accidents severity (Train set size: 135271, 18). To test the models I used 18 attributes for 33,818 observations of accidents severity (Test set size: 33818, 18).

k-Nearest Neighbors, Decision Tree, Random Forest, Logistic Regression, and Naive Bayes machine learning models were built and tuned.

Decision Tree, Random Forest and Naive Bayes demonstrated good results since the beginning. But Logistic Regression and especially k-Nearest Neighbors were generating wrong predictions (see Table 1), that was caused by the dataset limitations (the number of severe consequences of accidents is not enough to train those types of models). As the dataset is unbalanced therefore those 2 models were giving incorrect predictions for minority class 3 (severe injuries and fatalities).

5.1.2 Solution to the problems

I applied weights of each class when building the models. That mostly solved the problem of incorrect predictions (see average and weighted metrics in Table 1), except the kNN model (see models performance comparison in the Table 2).

Table 1. Average and weighted models performance comparison

Model	Precision		Recall		F1 Score	
	Macro average	Weighted average	Macro average	Weighted average	Macro average	Weighted average
Decision Tree	1.00	1.00	1.00	1.00	1.00	1.00
Random Forest	1.00	1.00	1.00	1.00	1.00	1.00
Naive Bayes	0.96	0.99	0.99	0.99	0.98	0.99
Logistic Regression	0.99	0.99	**0.89**	**0.99**	**0.93**	**0.99**
k-Nearest Neighbors	0.95	0.92	**0.61**	**0.92**	**0.61**	**0.91**

5.1.3 Performances of different models

During the performance assessment it is important to note that it is crucial to get correct predictions for the class 3, the most severe consequences of car accidents. As we already saw, it is fairly easy to predict property damage (22,114 accidents - majority cases) and even light injuries (11,103 accidents) using any model, but it takes some effort to get good prediction for severe injuries and fatalities (601 accidents - minority cases).

The comparison of models is presented in the Table 1.

Table 2. Performance comparison for machine learning classification models

Model	Metrics (total and for each severity class)					
	Accuracy	F1 Score	Precision	Recall	F1 Score	Confusion Matrix
Decision Tree	1.000	1.000	1 1.00 2 1.00 3 1.00	1 1.00 2 1.00 3 1.00	1 1.00 2 1.00 3 1.00	22114 0 0 0 11103 0 0 0 601
Random Forest	1.000	1.000	1 1.00 2 1.00 3 1.00	1 1.00 2 1.00 3 1.00	1 1.00 2 1.00 3 1.00	22114 0 0 0 11103 0 0 0 601
Naive Bayes	0.990	0.990	1 1.00 2 0.98 3 0.92	1 0.99 2 1.00 3 1.00	1 0.99 2 0.99 3 0.96	21820 280 14 0 11064 39 0 0 601
Logistic Regression	0.994	0.995	1 1.00 2 0.99 3 1.00	1 1.00 2 1.00 3 0.68	1 1.00 2 0.99 3 0.81	22114 0 0 0 11103 0 48 145 408
k-Nearest Neighbors	0.916	0.925	1 0.91 2 0.92 3 1.00	1 0.99 2 0.82 3 0.01	1 0.95 2 0.87 3 0.03	21847 267 0 1994 9109 0 99 494 8

6 RESULTS

Among the individual models, the Decision Tree model performed the best, not only because it showed 100% accuracy, but mostly because it predicted the most severe cases correctly. Random Forest performed the same, because it is simply a collection of decision trees whose results are aggregated into one final result. Random Forest could be useful in case Decision tree would not do the trick; but in this particular project Decision Tree classification model works perfectly fine on its own.

Naive Bayes prediction model showed 0.990 accuracy score, and although it had some errors in class 1 and 2, it was the second after the best result. This model was able to give correct prediction for all positive samples of minority class 3.

Logistic Regression prediction model showed 0.994 accuracy score, but its recall for severity class 3 was only 0.68. That means the classifier was able to find only 68% of the positive samples of minority class 3, therefore it's not good enough.

k-Nearest Neighbors prediction model showed 0.916 accuracy score and 0.925 F1 score for k=17, but that could be the problem for the minority of cases, accidents of class 3. Due to unbalanced dataset kNN model is not able to learn predicting class 3 correctly (recall for the class 3 is only 0.01, and F1 score is 0.03). kNN model performance was the worst in this case study, despite high overall accuracy.

7 CONCLUSIONS

In this case study, I analyzed the relationship between car accidents severity and different factors that contribute to consequences of the accidents. I identified environmental factors (weather, road conditions, and light conditions) and behavioral factors (driver's inattention, DUI, speeding and other driving rules violations) and geographical factors (latitude and longitude of location, type of road, particular part of the street and type of an accident itself in terms of its participants) among the most important features that affect a severity of an accident. Each of those factors alone is not enough to predict the outcome of an accident as there is no linear correlation between them and severity, so they should be used together to get god prediction results. That is the reason why decision tree model is the best one out of 5 classification models built to predict whether the accident would have severe consequences.

The decision tree model can be useful in helping city management in a number of ways. For example, it could help identify locations and factors of severe car accidents to adjust construction plans for road infrastructure and bridges, or to impose of traffic restrictions in the city center, or plan speed limits at some road parts.

8 DISCUSSION AND FUTURE DEVELOPMENT

I was able to achieve ~10% improvement for Logistic Regression model and ~30% improvement for kNN model using weighted samples. I got 100% accuracy with the best classification model (decision tree) and 92% accuracy with the worst ML model (kNN).

I have several recommendations to discuss:

1. Create and publish car accidents risks map, based on number of past accidents, number of people involved in them and accidents severity. This map could show different colors for the streets where drivers should be extra cautious and – for example - be used as a part of GPS navigation software for drivers. Or it could be used by city officials to develop new driving rules and restrictions in Seattle. For example, some streets in city center can be converted from the motor vehicles traffic zones to bike lanes and walking zones, like in Stockholm or Copenhagen.

2. Use the car accidents risk map to adjust existing speed limits in Seattle. Back in 2016 the Seattle instituted a 20 mph speed limit on residential streets of 20 mph. All other streets will be a default speed of 25 mph, though it may vary. Speeding is a $72 ticket. The speed limit on a street are published at the *Speed Limit Map*[1] on www.seattle.gov. For example, the speed limitations could undergo revision for locations of high accident risks. It is a fact that 9 out of 10 pedestrians survive being hit by a vehicle at 20 mph, at 30 mph only 5 out of 10 do, and at 40 mph 90% of people die from the impact.

3. Increase fines for distracted driving (particularly, for using electronics while driving). Since 2017 The Seattle Police can pull over and fine drivers for holding or using their cellphone or tablet while driving. The first ticket will cost at least $136, another ticket within 5 years will cost at least $234. This offence is called *Driving Under the Influence of Electronics*. I believe the fines in both cases could be higher, and their size should depend on the city area where the offence was registered: the fines have to be more severe for such offence in locations of high accident risks.

However, it would be useful to have extra data in the dataset and extract additional meaningful insights. For example, the data on drivers' demographics (age, sex, income level, marital status, education level, occupation, goal of the trip, etc.), or vehicle speed data at the time of accident, or daily traffic volume and speed limitations data at the locations of accidents, etc. This would help develop more detailed recommendations on city construction planning and traffic management.

[1] The link to the map http://www.seattle.gov/transportation/projects-and-programs/safety-first/vision-zero/speedlimits