ANALYSIS OF CAR ACCIDENTS IN SEATTLE

Miguel Cabello Reyes

21st of September, 2020

Table of contents

Introduction: Business Problem

The Introduction/Business Problem of my Capstone Project consists on a study about the severity of car accidents. The main aim of this project is to detect the principal causes of the crashes. With these data we want to take decision to improve safety in Seattle's roads.

To solve this problem we have data about some interesting topics. Some of data is useless. We are going to focus our analysis in the study of Road conditions, weather, etc. Also, we are going to discuss about the place where the accident occurs. Even is important to now the type of accident or, for example, in frontal crashes or that kind of accidents that are less usual, is important to know the hour and the week day cause is probably that some stuffs like alcohol could be one of the causes.

For this reason the main idea that is going to be discuss in the analysis is the location of the crashes to try to get a conclusion about the roads conditions of different hoods of Seattle and which factors are being multiplied by these conditions (example: frontal crashes, accidents in corners, etc.)

The objective of this analysis is to help Seattle city to reduce the number of accidents

Data

As I said previously my analysis will consists on a study of safety of the roads in Seattle. For this study we are going to use some parameters as: Road conditions, Weather, Location, Severity of Injury, Severity Code, Hour and Type of crash.

With this data we can group by the data by severity code and get a describe dataframe from each group. With this, we can see the most commons injuries for high severity accidents. Then we can get the accidents from each 'hood' creating a simple formula based on longotude and latitude of each accident. With this study we can get the most dangerous areas in the city.

Other objective is display the data with cluster in the Seattle map and with different colors depending on the severity code. A choropleth map is other great opportunity to this type of display. These analysis can be doing without taking care about the weather or with it. In that form we can see which zones are more affected by weather.

Another factor that I think that is interesting is the hour of the accident. With that we can predict in which hours the Seattle Police Department should reinforce the road controls and in which areas.

Data Preparing and Cleaning

The first step that we should do is prepare the data to do the analysis. To reach this we are going to create list only with the interesting paremeters and we are going to drop the rest.

The second step is to fill the gaps in the dataset with NaN values

The data that we get initially is the following one the we show in the image.

	SEVERITYCODE	X	Y	OBJECTID	ADDRTYPE	LOCATION	SEVERITYDESC	COLLISIONTYPE	PERSONCOUNT	PEDCOUNT	...
0	2	-122.323148	47.703140	1	Intersection	5TH AVE NE AND NE 103RD ST	Injury Collision	Angles	2	0	...
1	1	-122.347294	47.647172	2	Block	AURORA BR BETWEEN RAYE ST AND BRIDGE WAY N	Property Damage Only Collision	Sideswipe	2	0	...
2	1	-122.334540	47.607871	3	Block	4TH AVE BETWEEN SENECA ST AND UNIVERSITY ST	Property Damage Only Collision	Parked Car	4	0	...
3	1	-122.334803	47.604803	4	Block	2ND AVE BETWEEN MARION ST AND MADISON ST	Property Damage Only Collision	Other	3	0	...
4	2	-122.306426	47.545739	5	Intersection	SWIFT AVE S AND SWIFT AV OFF RP	Injury Collision	Angles	2	0	...
...	...	...	...	...	...	...	...	...	...	...	...
194668	2	-122.290826	47.565408	219543	Block	34TH AVE S BETWEEN S DAKOTA ST AND S GENESEE ST	Injury Collision	Head On	3	0	...
194669	1	-122.344526	47.690924	219544	Block	AURORA AVE N BETWEEN N 85TH ST AND N 86TH ST	Property Damage Only Collision	Rear Ended	2	0	...
194670	2	-122.306689	47.683047	219545	Intersection	20TH AVE NE AND NE 75TH ST	Injury Collision	Left Turn	3	0	...
194671	2	-122.355317	47.678734	219546	Intersection	GREENWOOD AVE N AND N 68TH ST	Injury Collision	Cycles	2	0	...
194672	1	-122.289360	47.611017	219547	Block	34TH AVE BETWEEN E MARION ST AND E SPRING ST	Property Damage Only Collision	Rear Ended	2	0	...

	JUNCTIONTYPE	INATTENTIONIND	UNDERINFL	WEATHER	ROADCOND	LIGHTCOND	PEDROWNOTGRNT	SPEEDING	ST_COLCODE	HITPARKEDCAR
0	At Intersection (intersection related)	NaN	N	Overcast	Wet	Daylight	NaN	NaN	10	N
1	Mid-Block (not related to intersection)	NaN	0	Raining	Wet	Dark - Street Lights On	NaN	NaN	11	N
2	Mid-Block (not related to intersection)	NaN	0	Overcast	Dry	Daylight	NaN	NaN	32	N
3	Mid-Block (not related to intersection)	NaN	N	Clear	Dry	Daylight	NaN	NaN	23	N
4	At Intersection (intersection related)	NaN	0	Raining	Wet	Daylight	NaN	NaN	10	N
...	...	...	...	...	...	...	...	...	...	...
194668	Mid-Block (not related to intersection)	NaN	N	Clear	Dry	Daylight	NaN	NaN	24	N
194669	Mid-Block (not related to intersection)	Y	N	Raining	Wet	Daylight	NaN	NaN	13	N
194670	At Intersection (intersection related)	NaN	N	Clear	Dry	Daylight	NaN	NaN	28	N
194671	At Intersection (intersection related)	NaN	N	Clear	Dry	Dusk	NaN	NaN	5	N
194672	Mid-Block (not related to intersection)	NaN	N	Clear	Wet	Daylight	NaN	NaN	14	N

194673 rows × 24 columns

Analysis of severity by groups

The sum of the data about the severity code is the next.

```
                        X          Y      OBJECTID  PERSONCOUNT  PEDCOUNT  \
SEVERITYCODE
1            -122.330722  47.618888  107655.87677     2.329348  0.005268
2            -122.330048  47.621058  110410.92782     2.714357  0.111896

              PEDCYLCOUNT  VEHCOUNT
SEVERITYCODE
1                0.004975  1.943312
2                0.083316  1.867928
count    194673.000000
mean          1.298901
std           0.457778
min           1.000000
25%           1.000000
50%           1.000000
75%           2.000000
max           2.000000
Name: SEVERITYCODE, dtype: float64
```

The first conclusion that we get is that all accidents are in a severity range between 1 to 2. That means that there are no register fatalities accidents in this data base. The code 1 corresponds to prop damage and 2 to injury. Knowing that, a good way to continue with the analysis is trying to separate between "important" accidents (code 2) and little accidents (code 1)

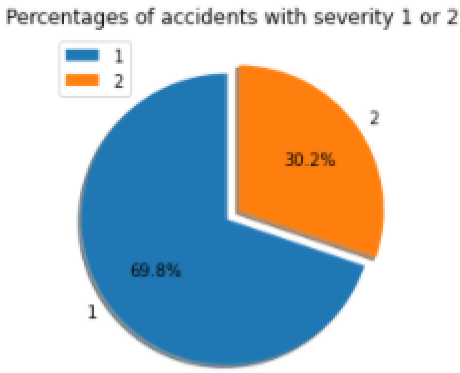
Other conclusion obtained if we watch at the mean values of vehicles, person, bicycles and pedestrians we can see how the accidents with higher severity has much more (in percentage) than little accidents. Dangerous accidents has more pedestrian and bicycles (are weaker than a car), has more people in each car (because if there are more people is most probably to have a person with worse injuries) and has less vehicles implicated in the accident, is a little difference but it could be because accidents with one car and one pedestrian and worse than accident between 2 cars. Dividng by groups of severity code:

SEVERITYCODE	X	Y	OBJECTID	INCKEY	COLDETKEY	REPORTNO	STATUS	ADDRTYPE	INTKEY	LOCATION	...	ROADCOND	LIGHTCOND	PEDROWNOTGRNT	SDOTCOLNUM	SPEEDING	ST_COLCODE	ST_COLDESC	SEGLANEKEY	CROSSWALKKEY	HITPARKEDCAR
1	132221	132221	136485	136485	136485	136485	136485	134750	37251	134081	...	132533	132405	460	81429	5802	136480	132622	136485	136485	136485
2	57118	57118	58188	58188	58188	58188	58188	57997	27819	57915	...	57128	57098	4207	33507	3531	58175	57147	58188	58188	58188

2 rows × 37 columns

In the percentage the result is the following:

```
SEVERITYCODE
1    69.83
2    30.17
Name: X, dtype: float64
```



Now we are going to study these severities inside each group. To carry on with this analysis, we show now the count of values of each one of the relevant columns in the DataFrame.

```
JUNCTIONTYPE
At Intersection (but not related to intersection)     1454
At Intersection (intersection related)               35420
Driveway Junction                                     7359
Mid-Block (but intersection related)                 15264
Mid-Block (not related to intersection)              68628
Ramp Junction                                           96
Unknown                                                  5
Name: X, dtype: int64
```

```
ROADCOND
Dry               82615
Ice                 911
Oil                  33
Other                78
Sand/Mud/Dirt        42
Snow/Slush          823
Standing Water       76
Unknown           13125
Wet               30689
Name: X, dtype: int64
```

```
WEATHER
Blowing Sand/Dirt              37
Clear                       73657
Fog/Smog/Smoke                369
Other                         663
Overcast                    18527
Partly Cloudy                   2
Raining                     21151
Severe Crosswind               17
Sleet/Hail/Freezing Rain       85
Snowing                       726
Unknown                     13115
Name: X, dtype: int64
```

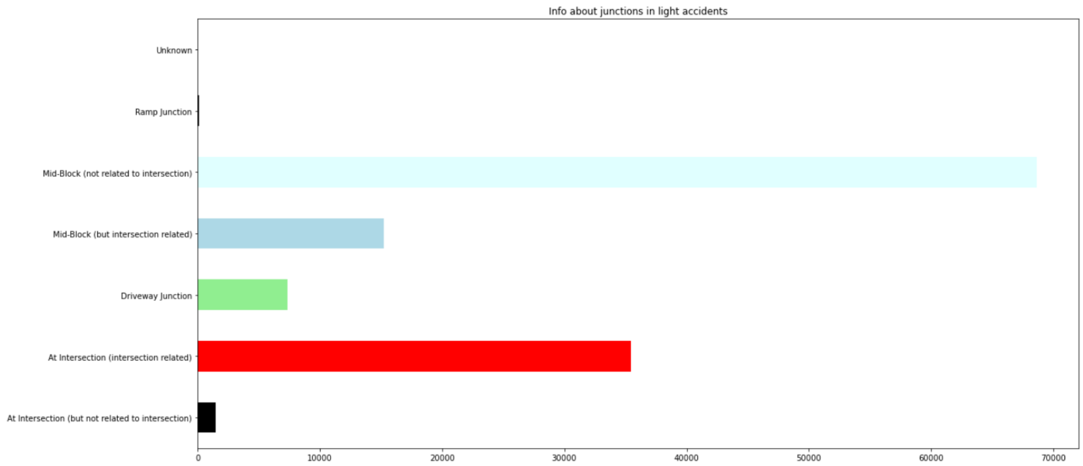
```
LIGHTCOND
Dark - No Street Lights      1132
Dark - Street Lights Off      846
Dark - Street Lights On     33122
Dark - Unknown Lighting         7
Dawn                         1612
Daylight                    75692
Dusk                         3858
Other                         151
Unknown                     11849
Name: X, dtype: int64
```

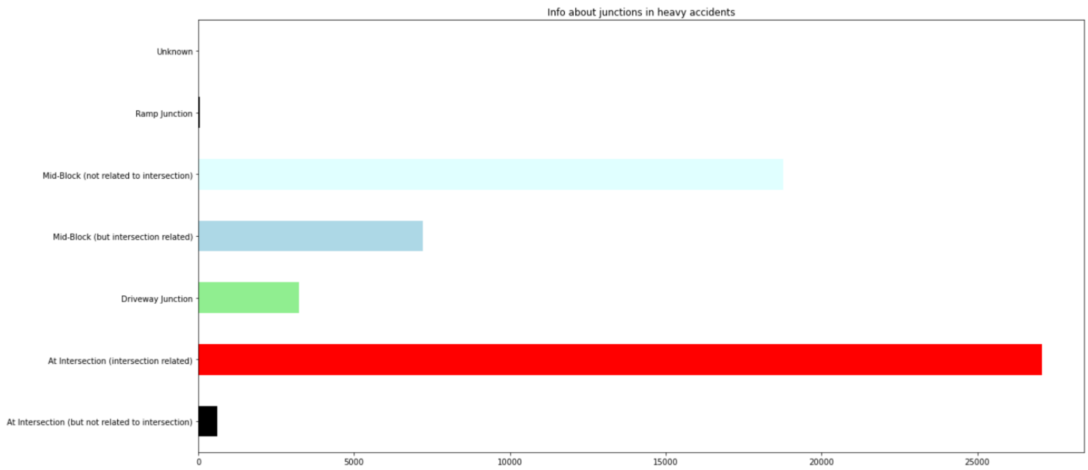
```
ADDRTYPE
Alley                0
Block            95191
Intersection     37030
Name: X, dtype: int64

SPEEDING
Y    5393
Name: X, dtype: int64
HITPARKEDCAR
N    125662
Y      6559
Name: X, dtype: int64
```

```
COLLISIONTYPE
Angles          20887
Cycles            666
Head On          1135
Left Turn        8242
Other           16481
Parked Car      43736
Pedestrian        670
Rear Ended      18749
Right Turn       2311
Sideswipe       15599
Name: X, dtype: int64
```

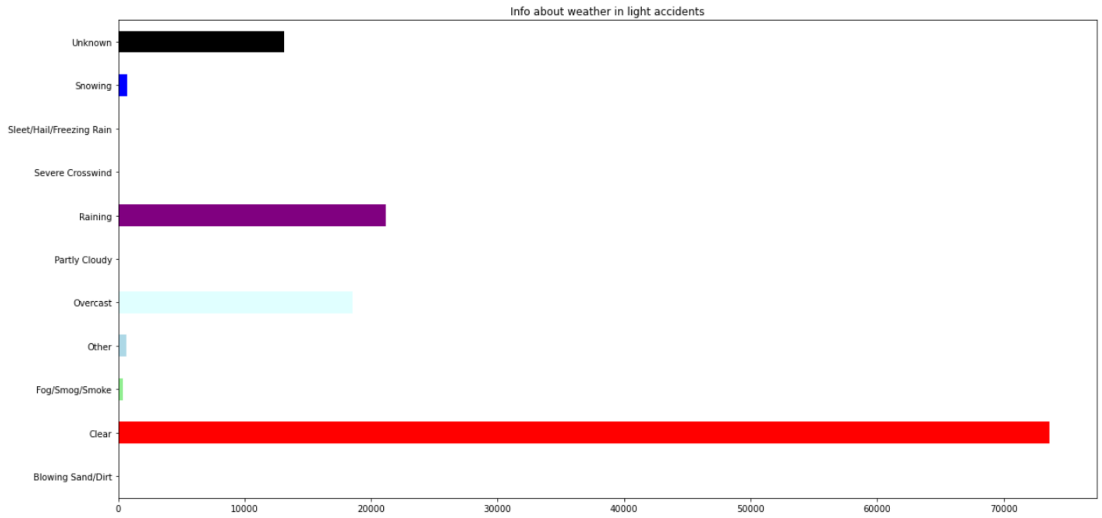
These data are more visual using horizontal bar plots. We choose the following ones and we think that is better doing this comparison with severity code 1 and severity code 2.

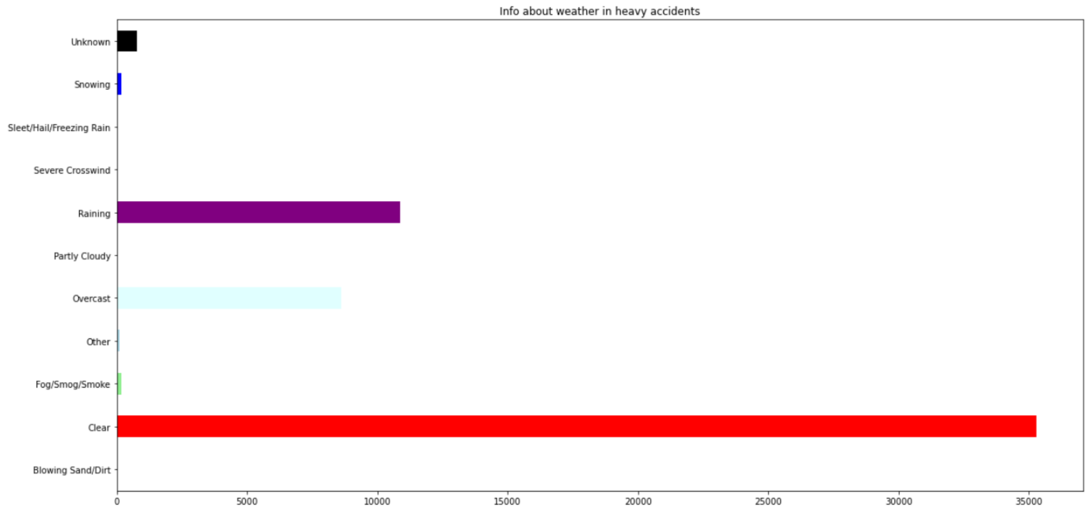
About junctions we can see that if the crash is in an intersection is more dangerous. And that the most typical scenarios are Mid-Block and Interjection.



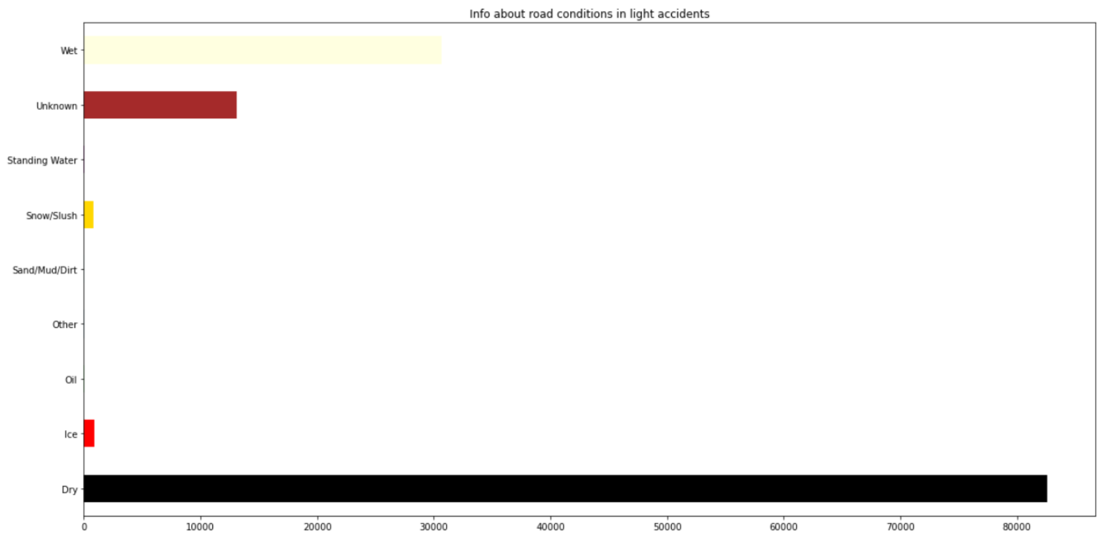


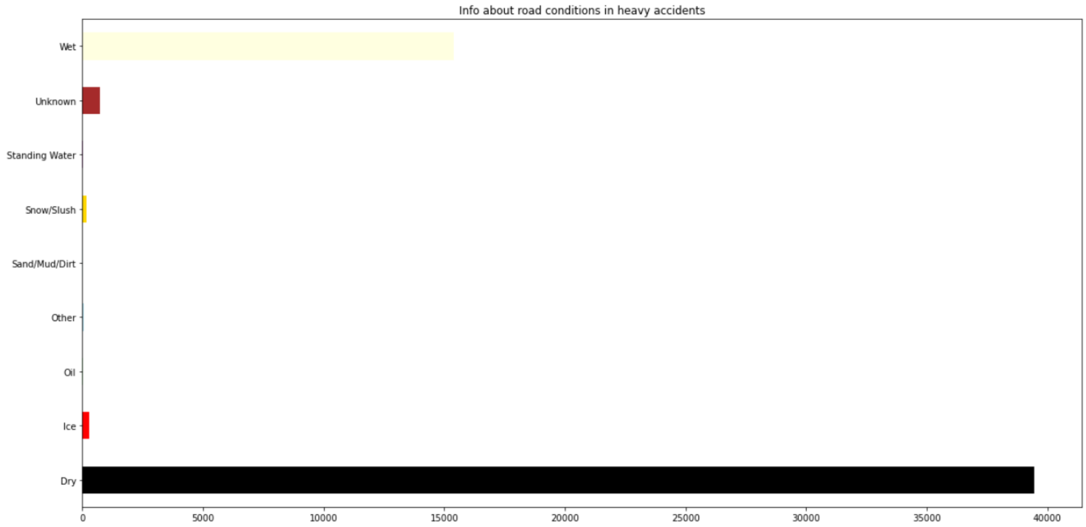
About weather we got the following, and we can think that weather is not an extremely parameter to our model:



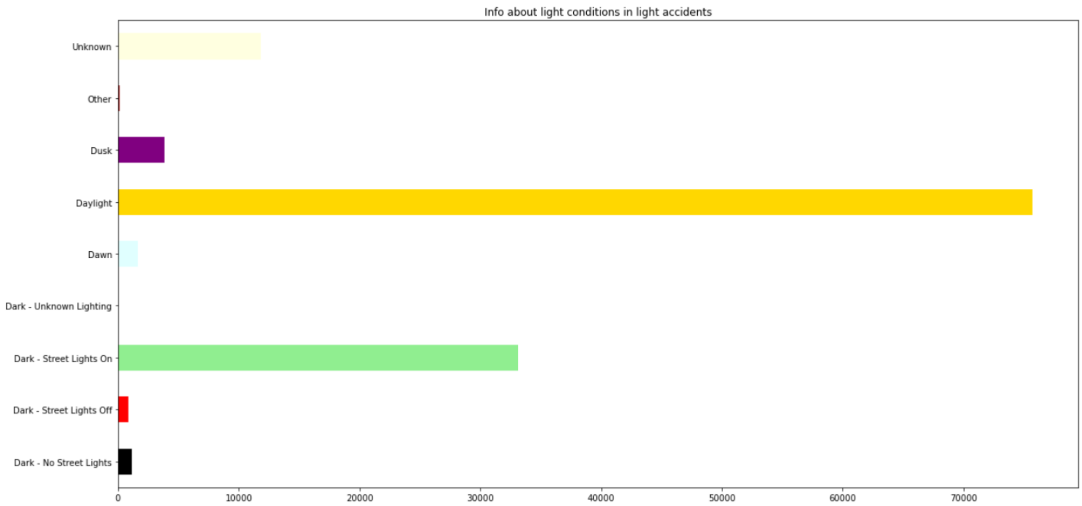


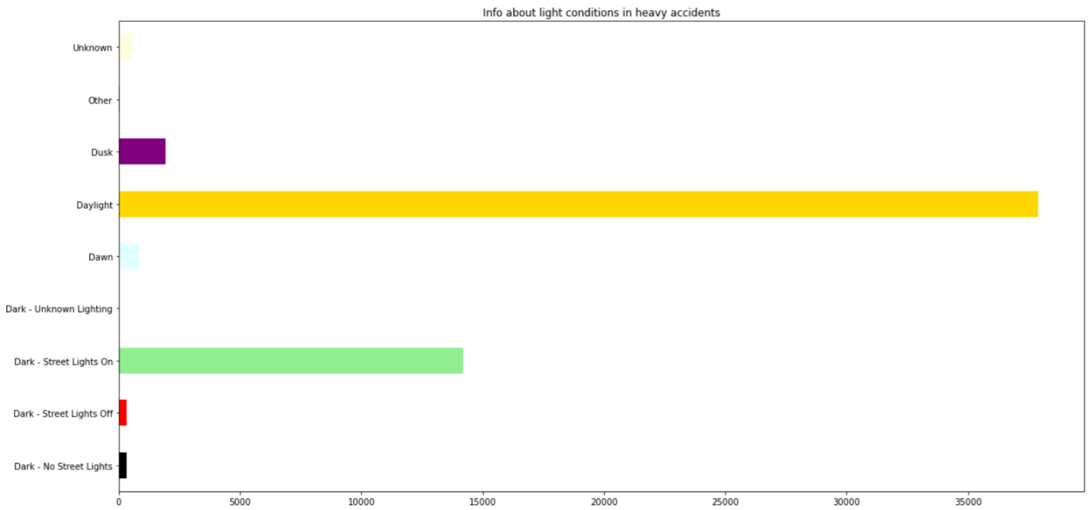
About the road conditions we can think that as weather is not an important parameter like weather.





About light conditions:





Light is not a parameter that show high differences between severity 1 and 2. The principal conclusion about this, is that causes of accidents in light accidents are more unknown that in heavy accidents.

All these graphics gave us an idea about the importance of weather conditions. The most usual values always is a good weather, but because is the normal situation. In fact, the relation between accidents with conditions as wet road is almost the half than with dry road, but to complete this analysis was be necessary to know the amount of days that rain in Seattle and probably we could see better the big influence of weather. The conclusion is that bad conditions that are more usual (for example: raining or wet road) are enough important to get them in count.

Model

Now we know more about accidents and the dataser, so we are going to create the model to predict the gravity of an injury.

We chose a multilinear model of regresion to predict de injury. The factors that we are going to take care are Pedestrians, Bycicles, Vehicles, Person in vehicle, RoadCondition, LightCondition, Speeding, Type of junction and parked car. With all these obviously will be a Multilinear Model.

Cause we have only 2 different posibilities (if severity is or 2) we need a high volume of data to the train group (80%) and we let the 20% of out dataset to test the model.

First of all, we are going to create a model only with the cuantitative variables: Pedestrians, Bycicles, Vehicles, Person in Vehicle

We obtain the next coeficients and interception:

```
Intercept:
 1.1185437010384098
Coefficients:
 [0.59633312 0.62483146 0.01292595 0.04730866]
```

So, the final ecuation for the model is:

$$Severity\ code = 1.11854 + Pedestrians \cdot 0.5963 + bicylce \cdot 0.6248 + vehicles \cdot 0.0129 + people \cdot 0.0473$$

The model works, but we should round the value that we obtain to get that is the severity code is 1 or 2. To fill the parameters we only need to know the number of person in a car, number of vehicles, number of pedestrians and number of bycicles.

The results that we obtain with the package sm.statsmodels.api:

```
                            OLS Regression Results
==============================================================================
Dep. Variable:           SEVERITYCODE   R-squared:                       0.130
Model:                            OLS   Adj. R-squared:                  0.130
Method:                 Least Squares   F-statistic:                     7258.
Date:                Sun, 20 Sep 2020   Prob (F-statistic):               0.00
Time:                        22:16:52   Log-Likelihood:            -1.1059e+05
No. Observations:              194673   AIC:                         2.212e+05
Df Residuals:                  194668   BIC:                         2.212e+05
Df Model:                           4
Covariance Type:            nonrobust
===============================================================================
                  coef    std err          t      P>|t|      [0.025      0.975]
-------------------------------------------------------------------------------
const           1.1185      0.003    323.685      0.000       1.112       1.125
PEDCOUNT        0.5963      0.005    116.883      0.000       0.586       0.606
PEDCYLCOUNT     0.6248      0.006    103.846      0.000       0.613       0.637
VEHCOUNT        0.0129      0.002      7.211      0.000       0.009       0.016
PERSONCOUNT     0.0473      0.001     60.457      0.000       0.046       0.049
==============================================================================
Omnibus:                    25605.630   Durbin-Watson:                   1.993
Prob(Omnibus):                  0.000   Jarque-Bera (JB):            31499.516
Skew:                           0.949   Prob(JB):                         0.00
Kurtosis:                       2.473   Cond. No.                         22.7
==============================================================================
```

With one example with:

Pedestrians = 0 Bicycles = 0 Vehicles = 2 People = 0

With the model we obtain a 1 of severity obviously.

The second example has the next parameters:

Pedestrians = 0 Bicycles = 1 Vehicles = 2 People = 0

We obtain a 2, because is most common that accidents with bycicles are more dangerous. If we add for example 3 pedestrians more, we obtain a severity of 4, but these means that we have a fatality accident.

The division in test and train group generate randomly these dataframes with 80% for train and 20 for test.

```
        PEDCOUNT  PEDCYLCOUNT  VEHCOUNT  PERSONCOUNT  SEVERITYCODE
84956          0            0         2            5             1
62977          1            0         1            3             2
36442          0            0         2            2             1
45504          0            0         2            4             1
10919          1            0         1            2             2
...          ...          ...       ...          ...           ...
57584          0            0         2            3             2
56210          0            0         2            2             1
76971          1            0         1            2             2
14277          0            0         2            2             1
52364          0            0         2            3             1

[155738 rows x 5 columns]
        PEDCOUNT  PEDCYLCOUNT  VEHCOUNT  PERSONCOUNT  SEVERITYCODE
2              0            0         3            4             1
10             0            0         2            2             1
21             0            0         3            5             2
27             0            0         2            2             1
30             0            0         2            3             1
...          ...          ...       ...          ...           ...
194650         0            0         5            5             2
194652         0            0         2            3             1
194659         0            0         2            2             1
194660         0            0         1            1             2
194662         0            0         2            2             1

[38935 rows x 5 columns]
```

Now we are going to add the qualitative parameters. First is substitute the values for numbers and we get the next dataframe.

	PEDCOUNT	PEDCYLCOUNT	VEHCOUNT	PERSONCOUNT	SEVERITYCODE	SPEEDING	HITPARKEDCAR	JUNCTIONTYPE	ROADCOND	LIGHTCOND
0	0	0	2	2	2	1	1	2	9	6
1	0	0	2	2	1	1	1	5	9	3
2	0	0	3	4	1	1	1	5	1	6
3	0	0	3	3	1	1	1	5	1	6
4	0	0	2	2	2	1	1	2	9	6
...	...	...	...	...	...	...	...	...	...	...
194668	0	0	2	3	2	1	1	5	1	6
194669	0	0	2	2	1	1	1	5	9	6
194670	0	0	2	3	2	1	1	2	1	6
194671	0	1	1	2	2	1	1	2	1	7
194672	0	0	2	2	1	1	1	5	9	6

194673 rows × 10 columns

And we get the next coefficients and intercept for this case.

```
Intercept:
 1.3380460569758668
Coefficients:
 [ 0.55271286  0.58464853  0.01677159  0.03988153  0.12175087 -0.12373325
 -0.04409615 -0.00274467 -0.00446666]
```

The final model is:

$$Severity\ code = 1.33804 + Pedestrians \cdot 0.5527 + bicylce \cdot 0.5846 + vehicles \cdot 0.0168 + people \cdot 0.0399 + speed\ cause \cdot 0.1218 + parked\ car \cdot (-0.1218) + junction \cdot (-0.0441) + road \cdot (-0.0027) + light \cdot (-0.0045)$$

With this data we obtain a 1:

Pedestrians = 0	Bicycles = 0	Vehicles = 2	People = 0
Speeding = 1	Road = 1	Junctions = 2	Light = 1

Parked Car = 1

With this statsmodels.api:

```
                            OLS Regression Results
==============================================================================
Dep. Variable:           SEVERITYCODE   R-squared:                       0.150
Model:                            OLS   Adj. R-squared:                  0.150
Method:                 Least Squares   F-statistic:                     3824.
Date:                Sun, 20 Sep 2020   Prob (F-statistic):               0.00
Time:                        23:10:04   Log-Likelihood:            -1.0827e+05
No. Observations:              194673   AIC:                         2.166e+05
Df Residuals:                  194663   BIC:                         2.167e+05
Df Model:                           9
Covariance Type:            nonrobust
================================================================================
                  coef    std err          t      P>|t|      [0.025      0.975]
--------------------------------------------------------------------------------
const           1.3225      0.008    155.646      0.000       1.306       1.339
PEDCOUNT        0.5871      0.005    115.628      0.000       0.577       0.597
PEDCYLCOUNT     0.6212      0.006    103.689      0.000       0.609       0.633
VEHCOUNT        0.0352      0.002     18.938      0.000       0.032       0.039
PERSONCOUNT     0.0405      0.001     51.769      0.000       0.039       0.042
SPEEDING        0.1248      0.005     27.564      0.000       0.116       0.134
HITPARKEDCAR   -0.1765      0.005    -34.651      0.000      -0.187      -0.167
JUNCTIONTYPE   -0.0289      0.001    -44.801      0.000      -0.030      -0.028
ROADCOND       -0.0048      0.000    -17.961      0.000      -0.005      -0.004
LIGHTCOND      -0.0108      0.001    -20.357      0.000      -0.012      -0.010
==============================================================================
Omnibus:                    24984.877   Durbin-Watson:                   1.990
Prob(Omnibus):                  0.000   Jarque-Bera (JB):            29117.008
Skew:                           0.904   Prob(JB):                         0.00
Kurtosis:                       2.437   Cond. No.                         84.6
==============================================================================
```

To understand that is important to know the code of numbers for parameters. Is the following:

road_conditions = {'Dry': 1,' Ice': 2, 'Oil': 3, 'Other': 4, 'Sand/Mud/Dirty': 5, 'Snow/Slush': 6, 'Standing Water': 7, 'Unknown': 8, 'Wet': 9, np.nan: 0, 'Sand/Mud/Dirt': 5}

parked_car = {'N': 1, 'Y': 2}

junction = {'At Intersection (but not related to intersection)': 1, 'At Intersection (intersection related)': 2, 'Driveway Junction': 3, 'Mid-Block (but intersection related)': 4, 'Mid-Block (not related to intersection)': 5, 'Ramp Junction': 6, 'Unknown': 7, np.nan: 0}

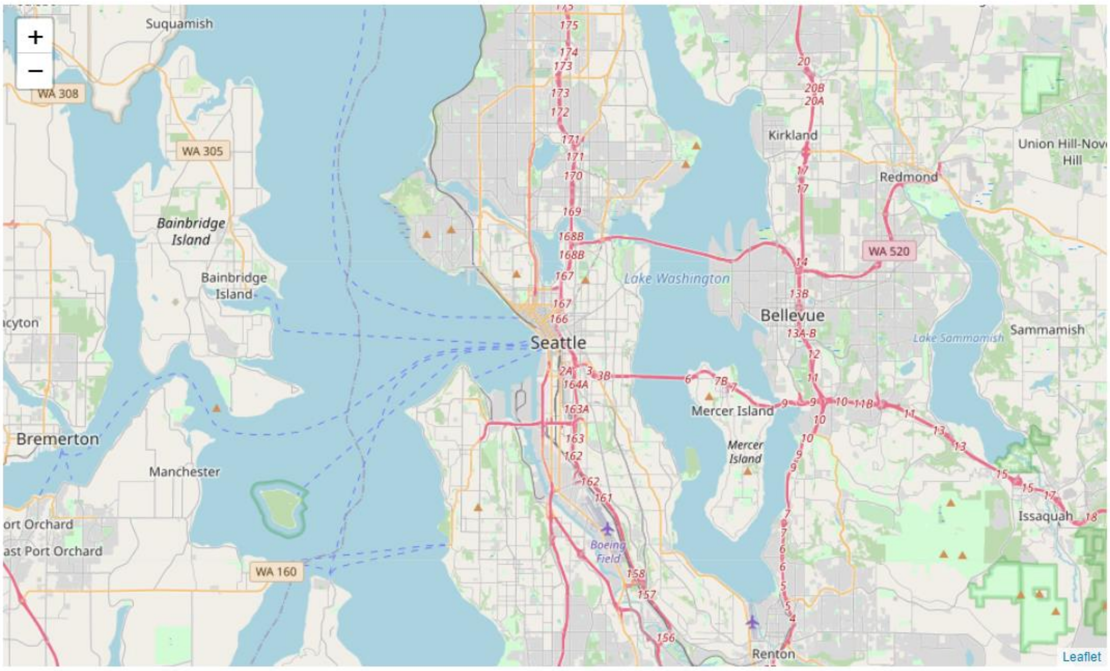
light_conditions = {'Dark - No Street Lights': 1, 'Dark - Street Lights Off': 2, 'Dark - Street Lights On': 3, 'Dark - Unknown Lighting': 4, 'Dawn': 5, 'Daylight': 6, 'Dusk': 7, 'Other': 8, 'Unknown': 9, np.nan: 0}

speeding = {np.nan: 1, ‘Y’: 2}

Finally, we are going to place all the accidents in a Map with Markers

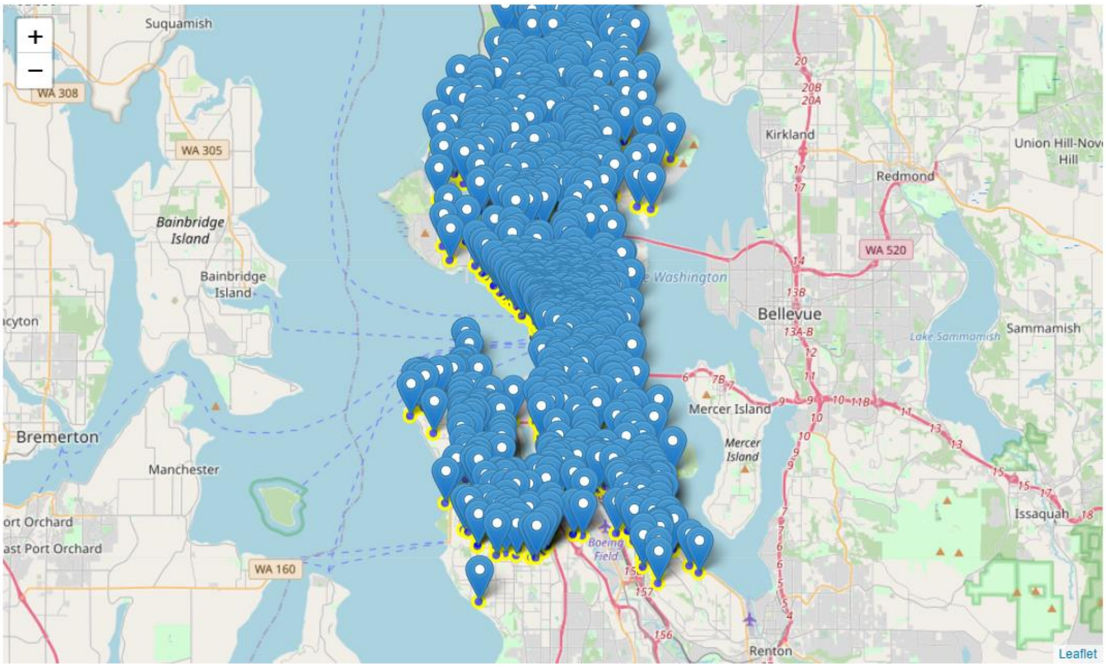
Maps

We generate with the library folium the map of Seattle city.



Now we add a marker from the first 1000 data in the dataframe

We obtain the following image.



To improve the map, we better use a Stamen Toner with the markers in clusters. With this we can see how the accidents are more frequently in the center and north of the city, and the most affected areas following the main road of the city (From North to South)

With this we can see how are spreaded the accidents over a toner map of Seattle

Results

The results of this studio are very bright. We could learn how all datasets has missed data and is important know how to deal with it. The main result to comment is that add values of weather, road conditions or light create an overfitting model, cause majority of accidents occurs with standard conditions. This make that these parameters haven't a big influence in the creation of the model.

The best model that I found is the model with the people and vehicles that are in the accident. Because to predict the danger is quite difficult to do it trying to find the causes of the accident but is quite easier if we analyze the potential people that could be injured.

We analyze the locations of accidents and we see how the majority are concentred in the main road of the city. This allowed us to get the conclusion that accidents are unusual in the hood areas. Are more usual in the city center.

Other result is that 1 of each 3 accidents is dangerous and that intersections represents the biggest danger about this topic.

Discussions

About the discussions I think that is interesting be careful with include all variables in a model cause of, you could get an overfitting model that doesn't represent reality. That happened if you add weather variables.

Another important thing is that plot variables is better in bar diagrams if there is to many groups inside this variable. If not, pie charts could be a fantastic option.

Subplots tools is a good option to compare plots. Allows to compare different variables faster only with a first seen. Probable are extremely useful for cases like this one, the use of maps (could be choropleth or maps with markers).

Probably could be a good option represent randomly some markers because in cases like this one that we have 150.000 rows in a dataset will do a huge time to calculate and a stacked map.

Is important to know that to use qualitative variables as weather conditions in a model is important to define a code to substitute strings for numbers. And a good way to see the efficient of a model is using the package of statsmodels.api.

Conclusion

With all this analysis we can take some conclusions. To summarize are there:

- The spread of accidents is irregular. We only plot in the map a randomly part of 1000 accidents. In the city-center is higher.
- Carrying on with these, one reason could be that is most common accidents in interceptions (are principal in the city center).
- The model has a higher dependence of how many people and of what type are in the crash.
- The conditions of the enviroment are important too, but less, with them we have over fitting in our model.
- Weather conditions are not good enough for our model, cause most of accidents occurs days with good weather. With the bar plots we can see the influence, but the big amount of sunny days with accidents make that when we add this data to the model, we have overfitting.