IBM Capstone Project

"Weather-related car crashes severity prediction"

Introduction/Business Problem

Problem

Official data for 2007-2016 provided by the Booz Allen Hamilton concluded that there are over 5,891,000 vehicle crashes each year on U.S., from where 1,235,145 are weather-related (approx. 21% of them) with an average of 418,005 persons injured and 5,376 persons killed [1].

From the 1,235,145 weather-related vehicle crashes, the majority occur on wet pavement with the 70% of them but only 46% during the rainfall, then the cases related with winter conditions being those during snow or sleet with only the 18% of them, this significant difference between the first causes and the rest could be because rain is a phenomenon present during all the year,

In comparison with the fatalities related with weather disasters or conditions, like tornados or heat, with an combined number of deaths of 379 persons per year, weather-related vehicle crashes kills 14.2 more times [2] and could have been avoided just by staying away from the streets during the weather phenomenon.

Proposed Solution

In order to reduce the number of weather-related vehicle crashes a Machine Learning model is proposed, a model capable of predict the severity of a possible collision if the person decides to drive, from property damage to fatality, aiming to make the person aware and possibly influence him to stay at home.

The deploy model ideally should work by asking the user the location where he/she would like to go, trace a road, identify the zones where the user would have driven, extract some data from those zones through internet, predict the severity of a possible collision in those zones and present the results to the user so he could take his decision about go out or stay away from the streets.

Stakeholders

Government: In 2009, the state of Washington presented an average management cost per crash of $125 for 3,880 crashes, which is the same to $485,000 per year in 2009 dollars [3] and an equivalent of $587,592 in 2020 dollars [4] just in Washington, from where the 21% would be for weather-related car crashes. With this application federal governments could take better prevention strategies per zones or give free access to citizens so they could be aware of the danger before going out to streets.

Insurers: Approximately the 50% of all motor vehicle crash costs are paid by private insurers, in 2013 [5]:

- The average auto liability claim for property damage was $3,231.
- The average auto liability claim for bodily injury was $15,443.
- The average collision claim was $3,144.
- The average comprehensive claim was $1,621.

Considering the 1,235,145 weather-related vehicle crashes we are talking about billions of 2020 dollars, with this application insurers could design new cheaper coverages where the client has the obligation to stay away from the roads if the severity of a possible car collision overcome some threshold.

Delivery and transportation services: Companies like Uber or Doordash could apply extra fees based on the severity of a possible weather-related car crash when an user ask for their services.

Data

Dataset

The dataset called "Collisions" was provided by Transportation_SeattleCityGIS and obtained from the gisdata.seattle.gov website [6], it contain all the collisions provided by SPD in Seattle and recorded by Traffic Record since 2004 and it is update weekly, at this moment it has 221,525 car crashes reports with 40 different attributes like [7]:

- Latitude and Longitude.
- Location (Name or description)
- Severity
- Collision Type
- Environment Conditions, etc.

Features and Labels

To predict the severity of a collision build a Machine Learning model was train based on attributes like:

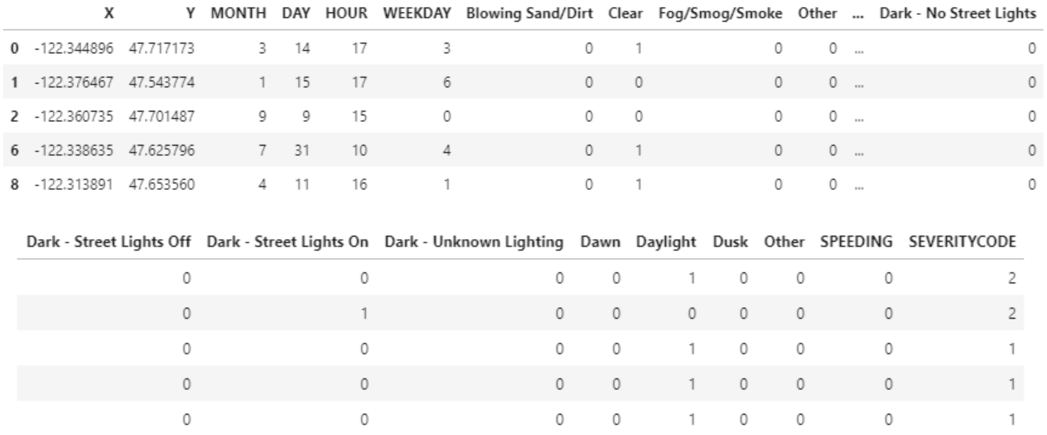
- The location where the person would transit (longitude, latitude, neighborhood)
- Date (Day and Month)
- Time (HH)
- Weather conditions (Clear, Raining, Overcast, Snowing, etc.)
- Road conditions (Dry, Wet, Slush, etc.)
- Light conditions (Daylight, Dark with streetlights, Dusk, etc.)
- Speeding

And it had the task to predict between one of four possible severity classes:

- 1 - prop damage
- 2 - injury
- 2b - serious injury
- 3 - fatality

Data Wrangling

All observations which have an "unknown" severity code, weather, road or light conditions were drop, the date was separated on day and month attributes and the time transformed into a 0-23 integer. Weather, road, and light conditions categories were separated as one-hot encoded attributes, finally the rest of attributes were drop.



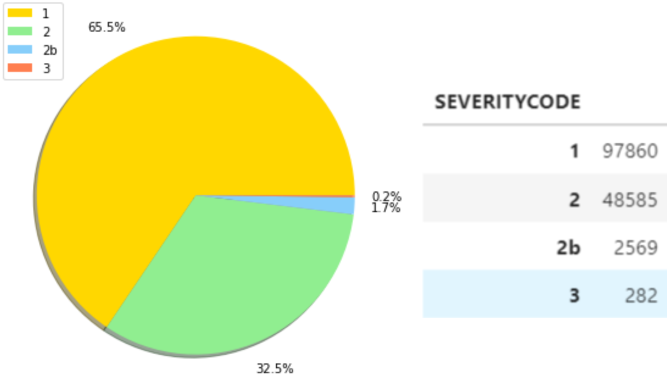
	X	Y	MONTH	DAY	HOUR	WEEKDAY	Blowing Sand/Dirt	Clear	Fog/Smog/Smoke	Other	...	Dark - No Street Lights	Dark - Street Lights Off	Dark - Street Lights On	Dark - Unknown Lighting	Dawn	Daylight	Dusk	Other	SPEEDING	SEVERITYCODE
0	-122.344896	47.717173	3	14	17	3	0	1	0	0	...	0	0	0	0	0	1	0	0	0	2
1	-122.376467	47.543774	1	15	17	6	0	0	0	0	...	0	0	1	0	0	0	0	0	0	2
2	-122.360735	47.701487	9	9	15	0	0	0	0	0	...	0	0	0	0	0	1	0	0	0	1
6	-122.338635	47.625796	7	31	10	4	0	1	0	0	...	0	0	0	0	0	1	0	0	0	1
8	-122.313891	47.653560	4	11	16	1	0	1	0	0	...	0	0	0	0	0	1	0	0	0	1

Fig.1 First five rows of the final dataset.

Methodology

Data Analysis

The final dataset consisted of 32 features with 149,296 records but the labels were highly imbalanced, the code 1-prop damage conform the 65% of the dataset, while the class 3-fatality only the 0.2% of it, this was fixed by undersampling class 1 and oversampling class 2b and 3 so the four classes had number of samples close to class 2 with 48,585.



SEVERITYCODE	
1	97860
2	48585
2b	2569
3	282

Fig.2 Sizes and proportions of each class.

But even with this imbalance, the four classes present very similar distributions on most of the features, observing the time of the accidents by each class on Fig.3 it can be easily observe that for the firsts three classes, the number of crashes increase along with the hours, with a peak around 17:00 hours and then start to decrease, but the class 3-fatality has also a relative high number of crashes from 00:00 to 02:00 hours.

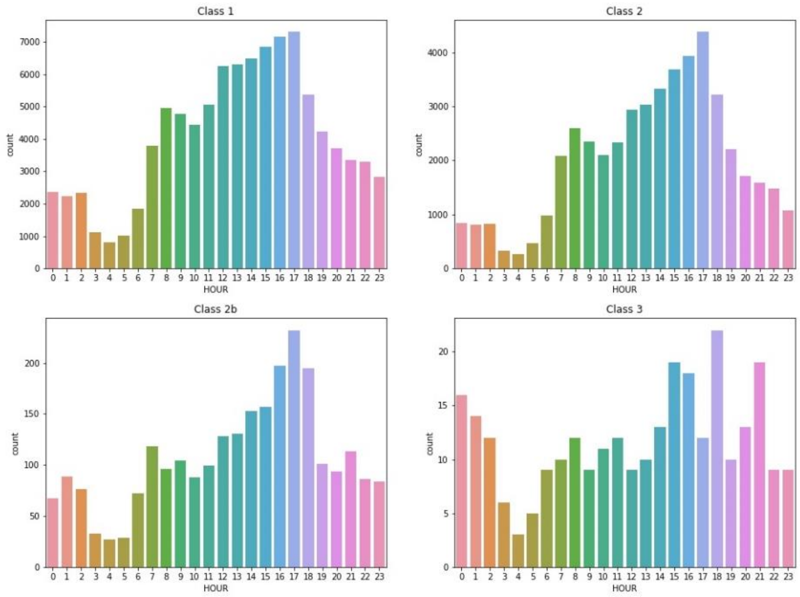


Fig.3 Number of crashes per hour for each class.

On Fig. 4 the number of crashes per day of the month is shown, for the firsts three classes the number of crashes increase around 76% almost instantly from day 10 onward, while the class 3-fatality present a more erratic behavior.

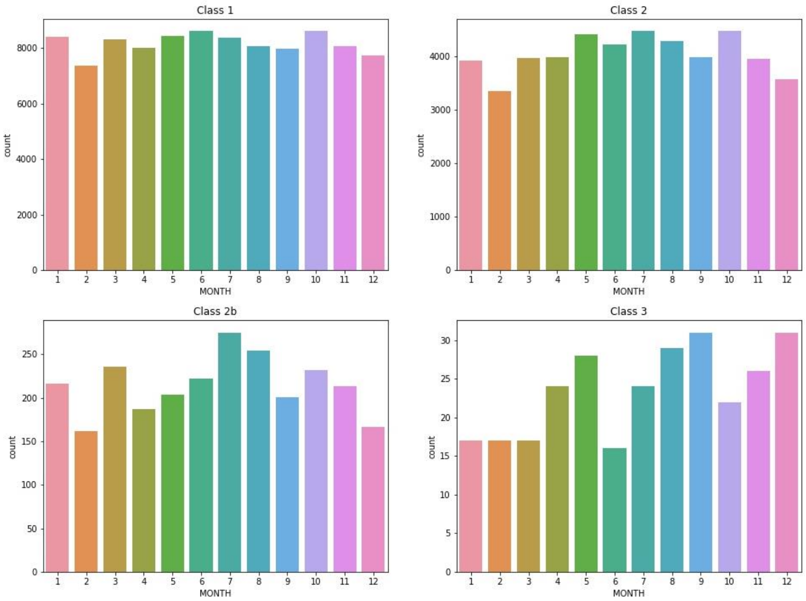


Fig.4 Number of crashes per day of the month for each class.

In the case of weather conditions, the four classes present similar proportions with the majority of the crashes during a *clear* weather and then on *overcast* or *raining* weather, and almost null cases for the rest of weather conditions, as it can be seen of Fig. 5.

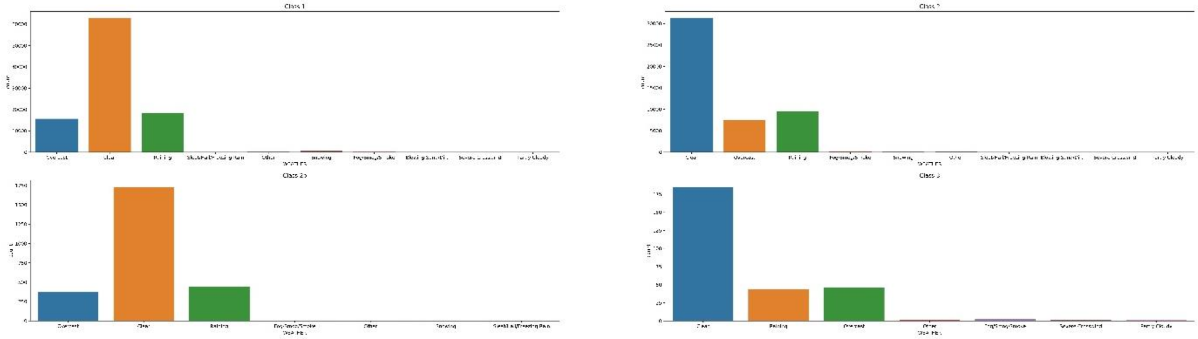


Fig.5 Number of crashes per weather for each class.

Something similar to weather conditions occur with the light conditions, with the four classes having the most of crashes during *daylight* and then during *dark – street lights on* but, as the severity of the accident increase, the proportion of accidents during *dark – street lights* respect to *daylight* conditions do as well, going from the 43% to 83%.

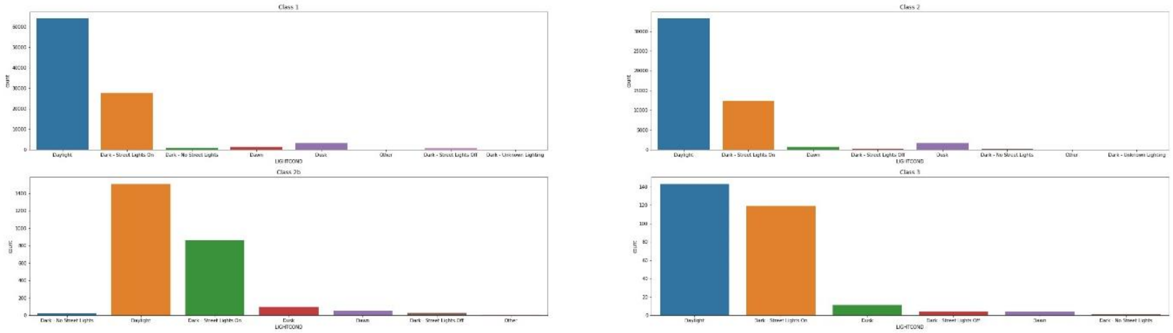


Fig.6 Number of crashes per light condition for each class.

Fig 7 shown the location on the map of Seattle of a small subset of 100 accidents per class: 1-prop damage (green), 2-injury (yellow), 2b-several injury (red) and 3-fatality (black), as it can be seen, none of the classes are isolate to some region or area of the city, but there is a concentration of accidents around Westlake, University Street and Pioneer Square.

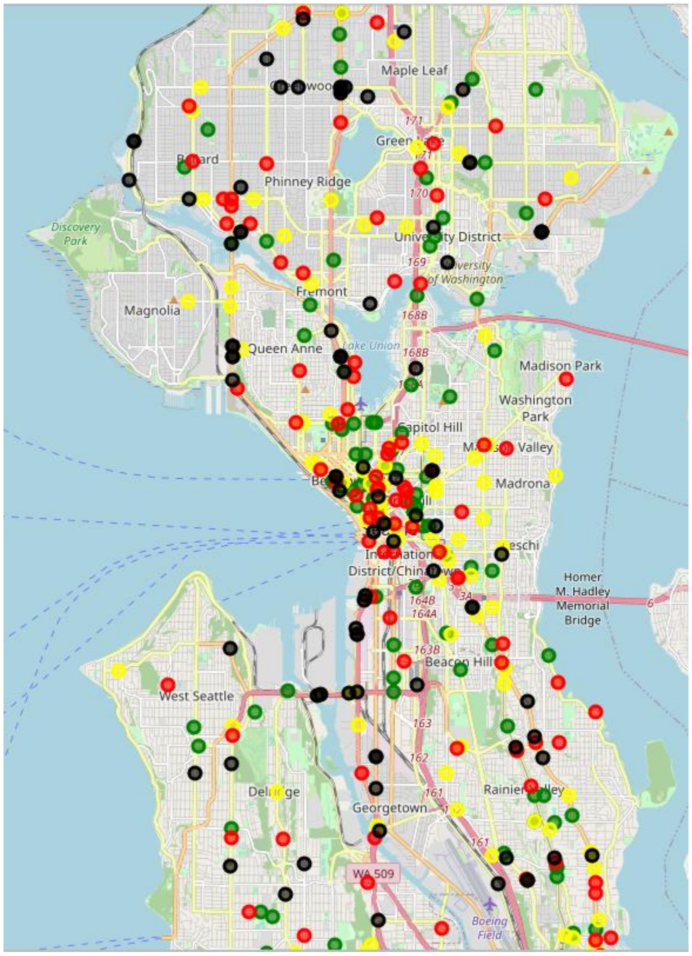


Fig.7 Map of Seattle with the location of 400 car accidents.

ML Models

The models were trained and evaluated trying different hyper parameters combinations and keep the one which achieve the highest evaluation F1 score, the final hyper parameters for those models were:

Decision Tree: A maximum depth of 40 levels, with the balanced option for the class_weight parameter so the model to took in consideration the unbalance on the data, and the entropy criterion for the attributes selection.

Random Forest: Also, the entropy criterion for the attributes selection with the balanced option for the class_weight parameter and 15 trees on the forest.

Logistic Regression: The algorithm was train for the fifteen possible combinations using the five solver options ('newton-cg', 'lbfgs', 'liblinear', 'sag', 'saga') and eight C coefficients (from 0.1 to 0.00000001), the best model was selected considering the test F1 score and training accuracy was selected, being the one using *liblinear* with a C equal to *0.01*.

Results

Observing the firsts nodes from the tree it can be analyze the features that the algorithm chooses as the ones with the biggest distinction impact on the data, as it is shown on Fig. 8,

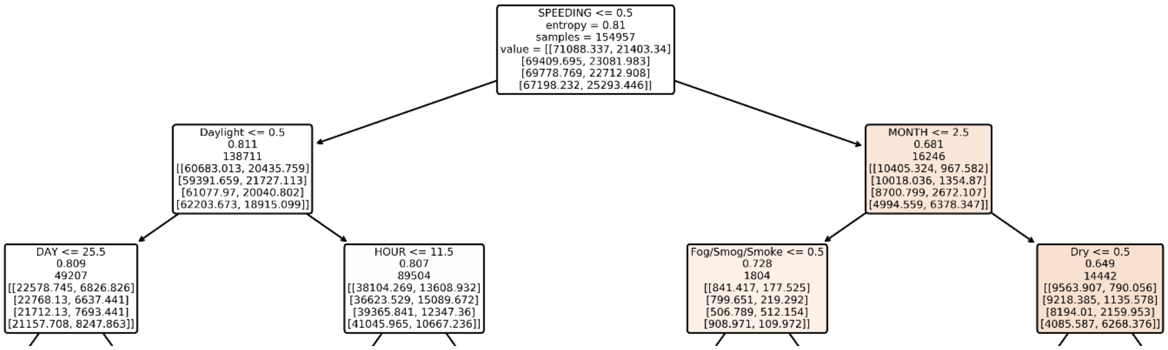


Fig.8 First three levels from the decision tree.

Fig. 9 shows the F1 score results obtained from all the linear regression trained models, except the one using the *liblinear* solver, all models presented almost identical results. The same occurs on Fig. 10 with the results from the training accuracy, and as the F1 score decrease, the training accuracy increase, except for the *liblinear* solver which both results increase as *C* also do.

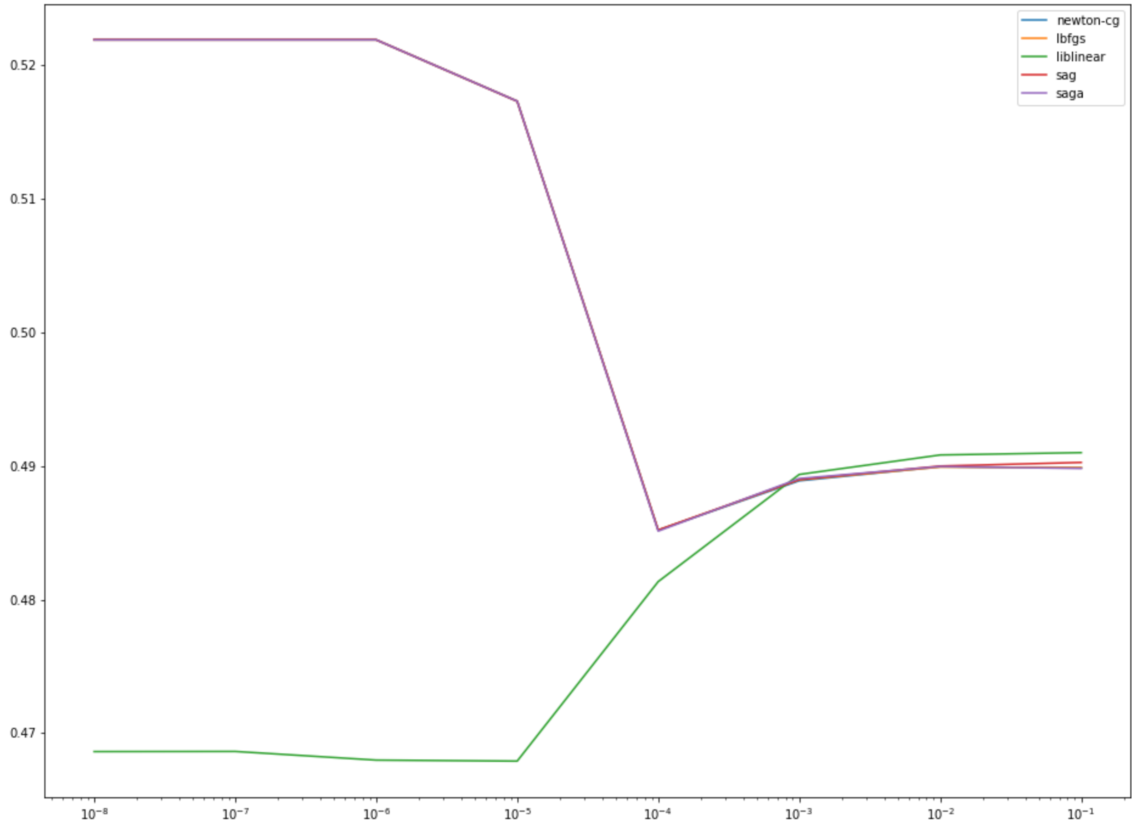


Fig.9 F1 score results from the linear regression models

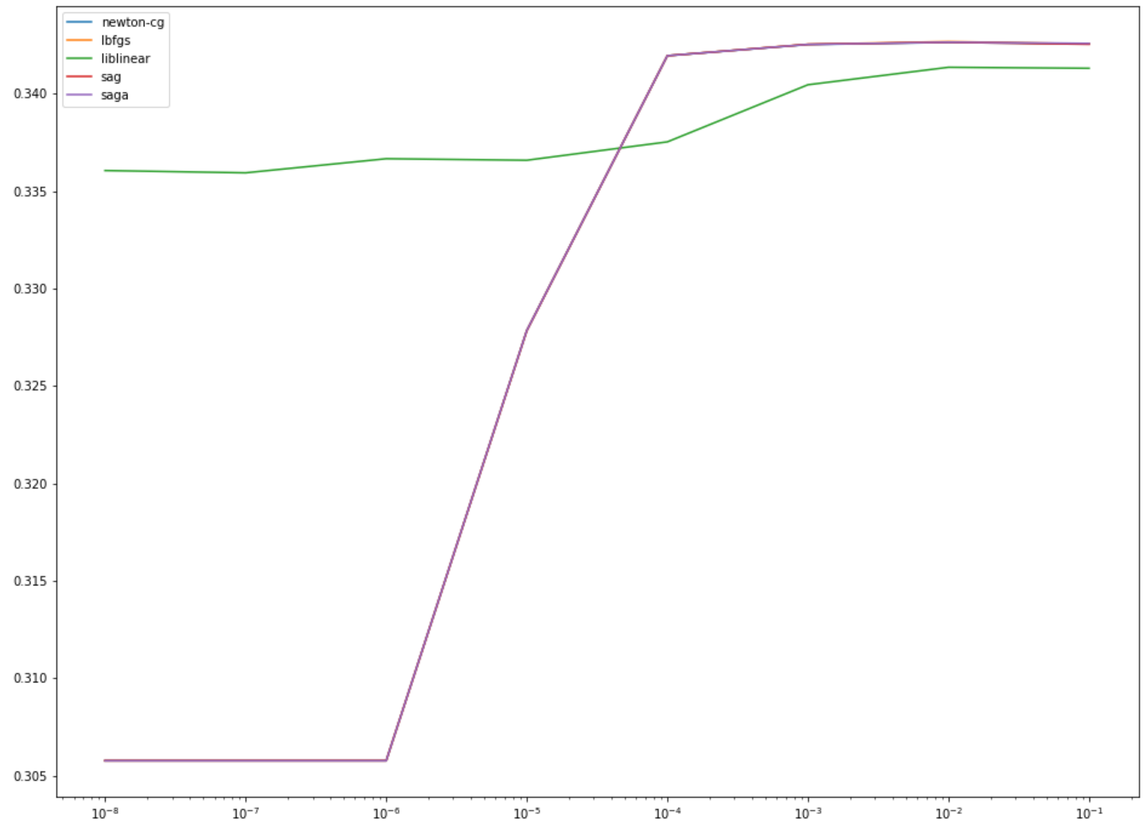


Fig.10 Training accuracy results from the linear regression models

Tab. 1 Comparison of the best results obtained from the trained models.

Model	Decision Tree	Random Forest	Logistic Regression
Training Accuracy	0.9996	0.9970	0.3414
Testing Accuracy	0.5027	0.5009	0.5483
Test F1 Score	0.52	0.53	0.49

Conclusions

The Logistic Regression model obtained the highest testing accuracy with a 55%, overcoming the rest of the models with a difference of 5%, while the Random Forest obtained the highest testing F1 score with a 53% and Logistic Regression de lowest with a 49%.

The imbalance on the data and the attributes on the dataset didn't provide enough information so the models could learnt to correctly classify the severity of a car crash on a commercialization level, however all the models achieve a testing accuracy above 50%, so they can be used as a first opinion.

In order to improve the model´s performance new features can be explored, like a *risk_neighbor_level attribute*, using a Geojson file with the neighbors of the city, the coordinates of the crashes and the severity of the crash, cluster the neighbors into 3-5 severity levels.

Also, new datasets from different cities can be used to increase the size of low number of crashes classes like 2b-several injury and 3-fatality.

Bibliography

[1] **How Do Weather Events Impact Roads?** https://ops.fhwa.dot.gov/weather/q1_roadimpact.htm
[2] **Weather-Related Vehicle Accidents Far More Deadly Than Tornadoes, Hurricanes, Floods** https://weather.com/safety/winter/news/weather-fatalities-car-crashes-accidents-united-states
[3] **Costs of Crashes to Government, United States, 2008** https://www.ncbi.nlm.nih.gov/pmc/articles/PMC3256813/
[4] **Dollar devaluation calculator** https://www.in2013dollars.com/us/inflation/2009?amount=485000
[5]**Cost of Auto Crashes & Statistics** http://www.rmiia.org/auto/traffic_safety/Cost_of_crashes.asp
[6] **Collisions (dataset source)**https://data-seattlecitygis.opendata.arcgis.com/datasets/collisions
[7] **Collisions-All Years (dataset metadata)** https://s3.us.cloud-object-storage.appdomain.cloud/cf-courses-data/CognitiveClass/DP0701EN/version-2/Metadata.pdf