# Machine Learning Engineer Nanodegree
Capstone Project

Leticia Portella
September 20th, 2018

# I. Definition

## Project Overview

Roads are considered one of the most common ways to travel and transport material in Brazil. Nevertheless, Brazilian roads are considered extremely dangerous and unsafe. According to OMS, Brazil is the fifth country with deaths in traffic accidents, with more than 47 thousand deaths per year.

There are three main types of road in Brazil: local, regional (by state) and national roads. National roads are monitored by the Federal Highway Police (Polícia Rodoviária Federal, aka, PRF), which keeps records on time, place and type of accidents that happened in these highways per year. The dataset are publicly available on the PRF website.

This project used the data from the PRF in an attempt to predict the type of victims an accident can have based on the characteristics of the highway, the moment of the accident and main characteristics of the accident. The main goal is to define if that accident will likely have no victims, injuries victims or death victims.

## Problem Statement

The main goal is try to predict whether an accident will have no victims, injured victims or dead victims. We considered information on the time of the accident (moment of the day, day of week, etc), on the place the accident happened (road number, kilometer, state, etc) and the accident characteristics (such as cause of the accident, and type of accident).

If we can predict, with a certain amount of confidence, the type of victims based solely on the local and time of the accident, one can assume that there is a problem on Brazilian Roads, and one can predict most dangerous areas. Even if this is not possible, the predictions could help identifying most problematic areas and help on prevention measures.

One possible approach if this 3 class scenario is not viable, is to create a model identifying if an accident will have victims (deaths or injured) or not or if an accident will have dead victims or not.

## Metrics

The main metrics used throughout the project was the accuracy score, which can handle multi class classification problems. The formula considered by Sklearn is the following:

$$\texttt{accuracy}(y, \hat{y}) = \frac{1}{n_{\text{samples}}} \sum_{i=0}^{n_{\text{samples}}-1} 1(\hat{y}_i = y_i)$$

I also used the classification report for detailed observation of recall and precision for each class.

For a better visualisation of the model performance, I used the confusion matrix_to visualize the results, using the script suggested on this page.

It is my believe that this three metrics together, gives a good ideia of where the model is failing, where is it doing it right and how can we improve it. Nevertheless, I always prioritise the **recall** metrics instead of any other, since for the purpose of this study, we want the minimum number of false negatives.

# II. Analysis

## Data Exploration

The datasets contained the following features:

| Feature Name | Feature Meaning | Feature Type |
|---|---|---|
| Id | accident identification | - |
| data_inversa | accident date | Categorical |
| dia_semana | weekday of the accident | Categorical |
| horario | accident hour | - |
| uf | state | Categorical |
| br | highway number | Categorical |
| km | Kilometer | Numerical |
| municipio | city | Categorical |
| causa_acidente | accident cause | Categorical |
| tipo_acidente | accident type | Categorical |
| classificacao_acidente | accident classification (target variable) | Categorical |
| fase_dia | moment of the day | Categorical |
| sentido_via | road way | Categorical |
| condicao_meteorologica | Climate | Categorical |
| tipo_pista | road type | Categorical |
| tracado_via | road layout | Categorical |

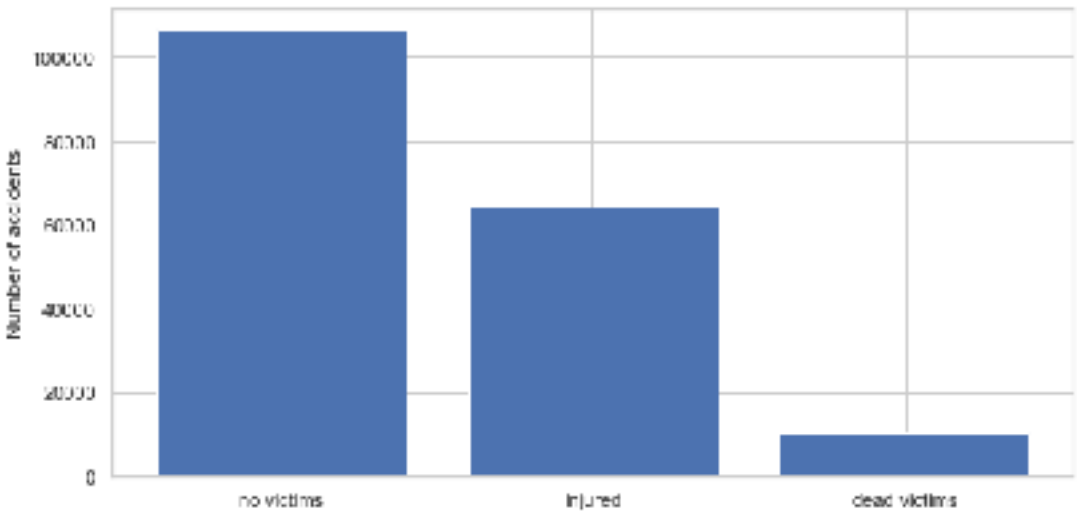
| Feature Name | Feature Meaning | Feature Type |
|---|---|---|
| uso_solo | soil use | Categorical |
| pessoas | Number of people | Numerical |
| mortos | Number of deaths | Numerical |
| feridos_leves | Number of light injured | Numerical |
| feridos_graves | Number of severe injured | Numerical |
| ilesos | Number of people not harmed | Numerical |
| ignorados | Ignored | Numerical |
| feridos | Total number of injured | Numerical |
| veiculos | Number of cars | Numerical |
| Latitude | Latitude | Numerical |
| Longitude | Longitude | Numerical |

The datasets from 2016 and 2017 had inconsistencies so a cleanup was necessary. The common problem found was that the categorical features had differences on names. On the cleanup, the names of categorical features were standardised and translated to english. The target variable, *classificacao_acidente*, was changed to values from 0 to 2 (no victims, injured victims and dead victims, respectively) and renamed as *target*.

Records that did not have information on moment of day, climate or had no information on the target variable, were removed. From a initial total of 184225 records, 180991 records were available after the cleanups.

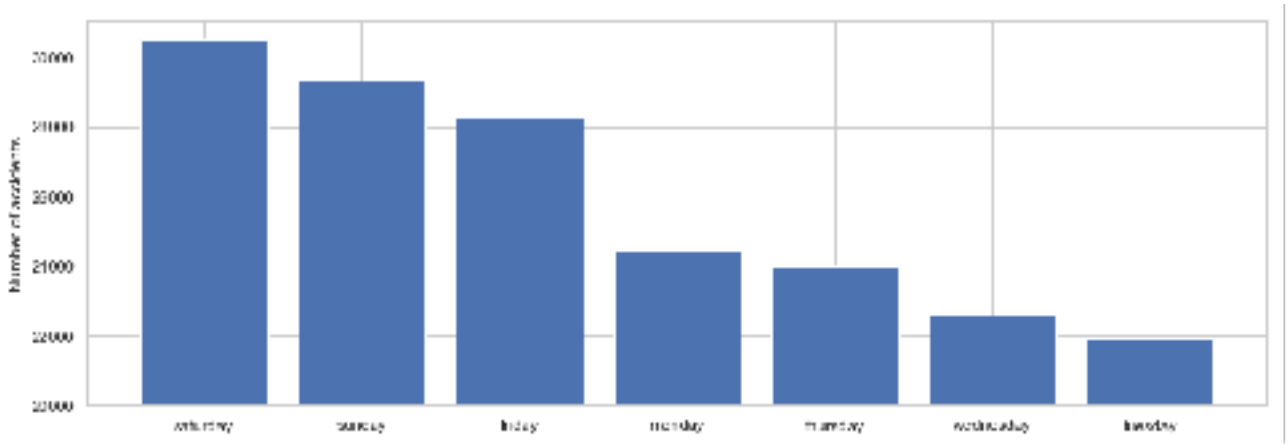
All cleanups were made on a single notebook and the cleaned dataset was exported and this was the dataset used on all analysis.

From all the 180991 records, 58.8% had no victims, 35.4% had injured victims and only 5.6% had dead victims. So we have a clear case of unbalanced classes:

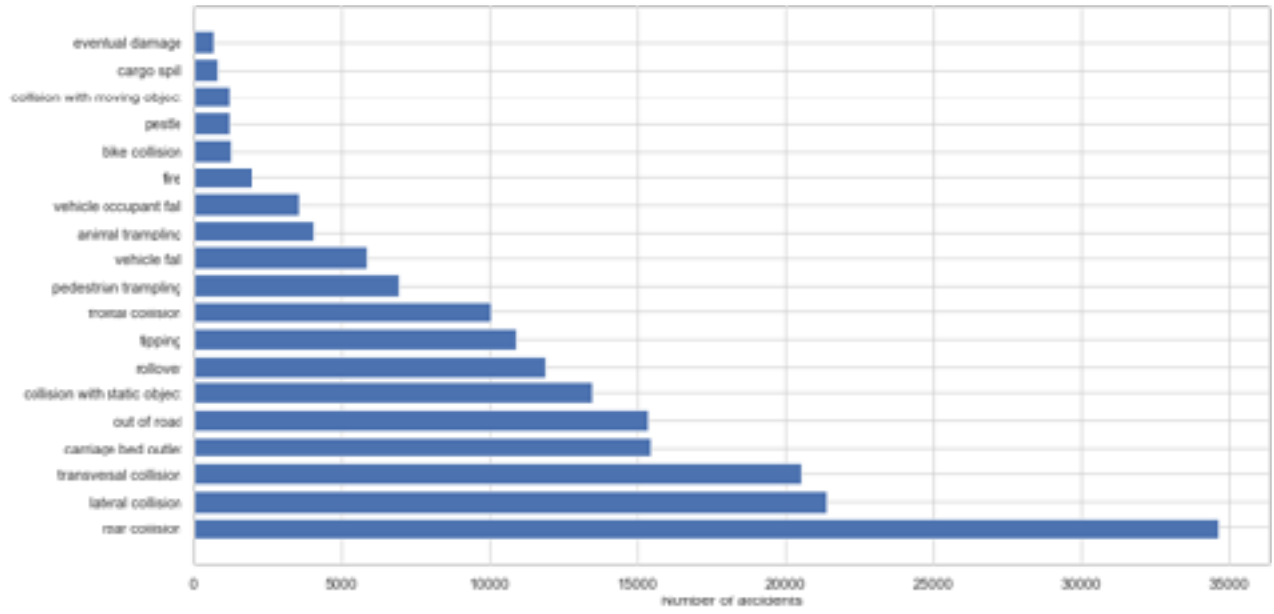


## Exploratory Visualization

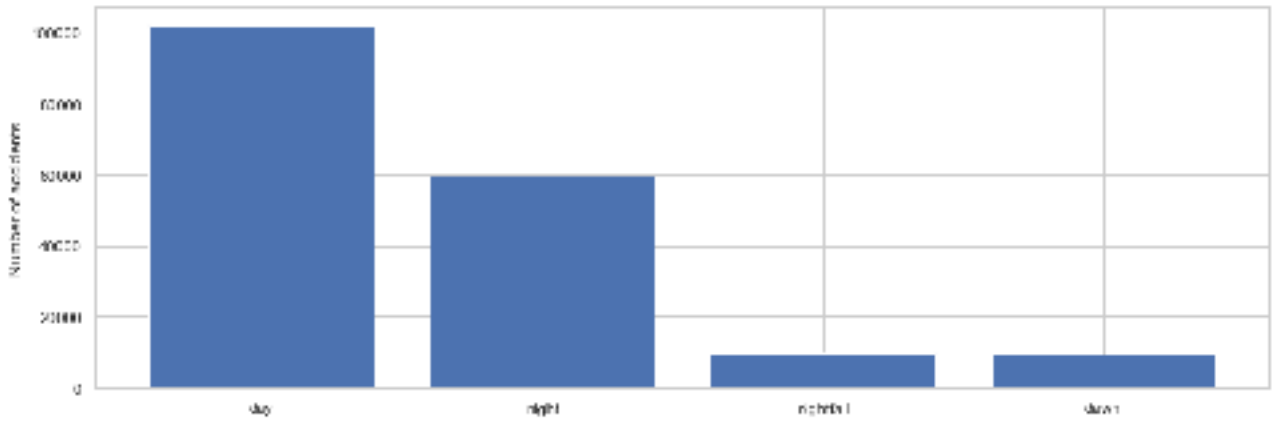
Most of the accidents happens during the weekends, being Saturday the day with higher number of accidents. Monday is the weekday with higher number of accidents, while Tuesday is the day with less number of accidents:



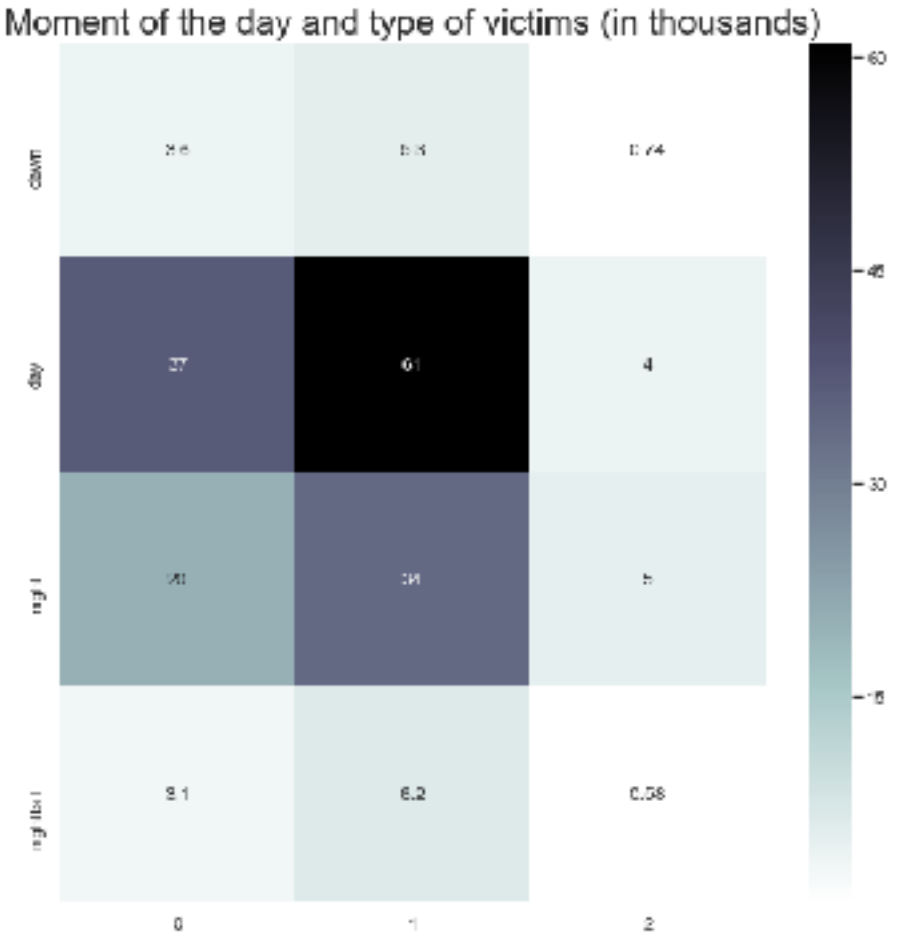
Rear collision are the most common type of accident, with almost 35000 accidents registered. Lateral and transversal collision are the second and third more common cause of accidents, with more than 20000 records.



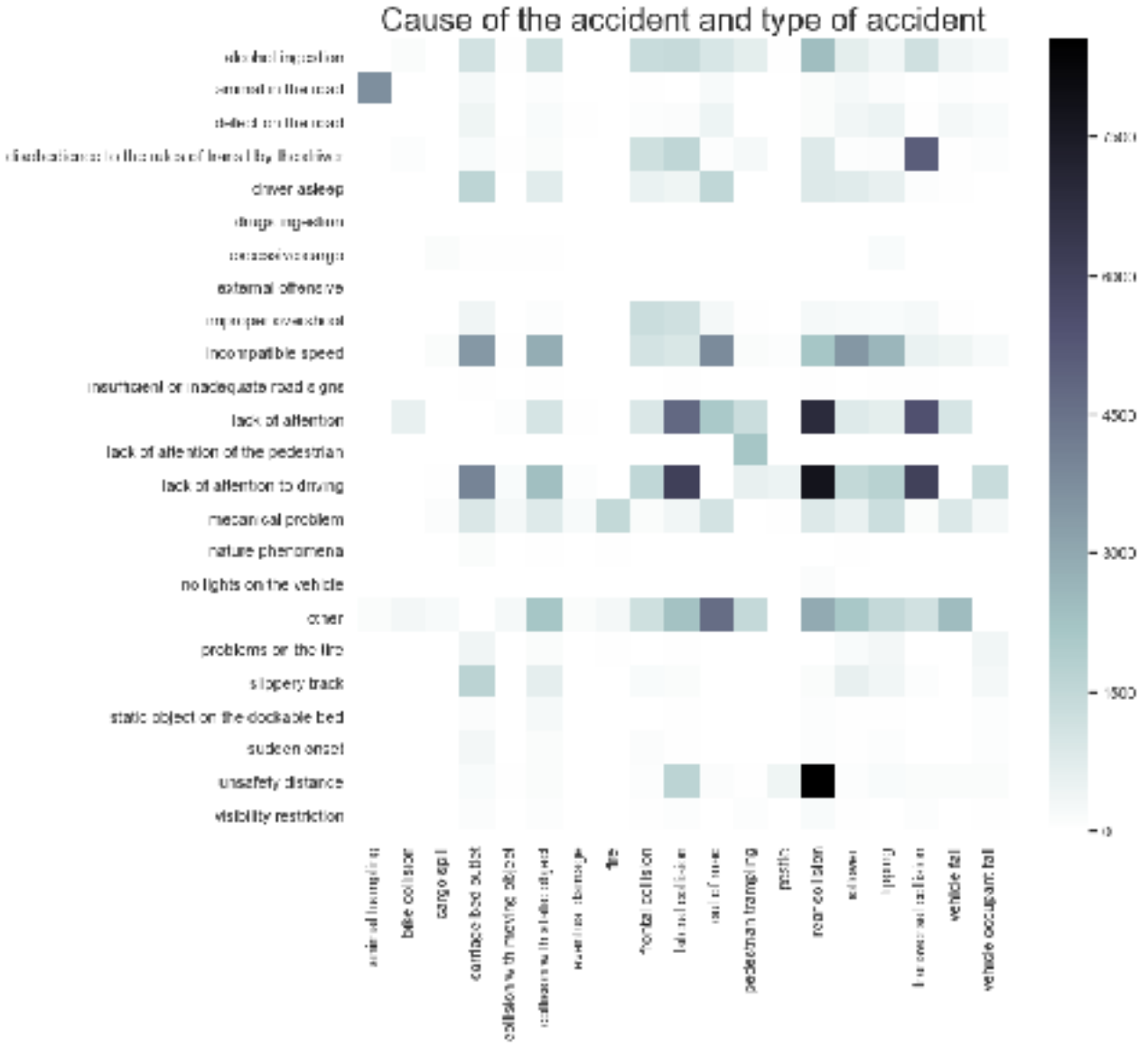

Most accidents happened during the day (56.3%), while a third happened at night (32.8%) and ~11% happened in the transition between day and night.



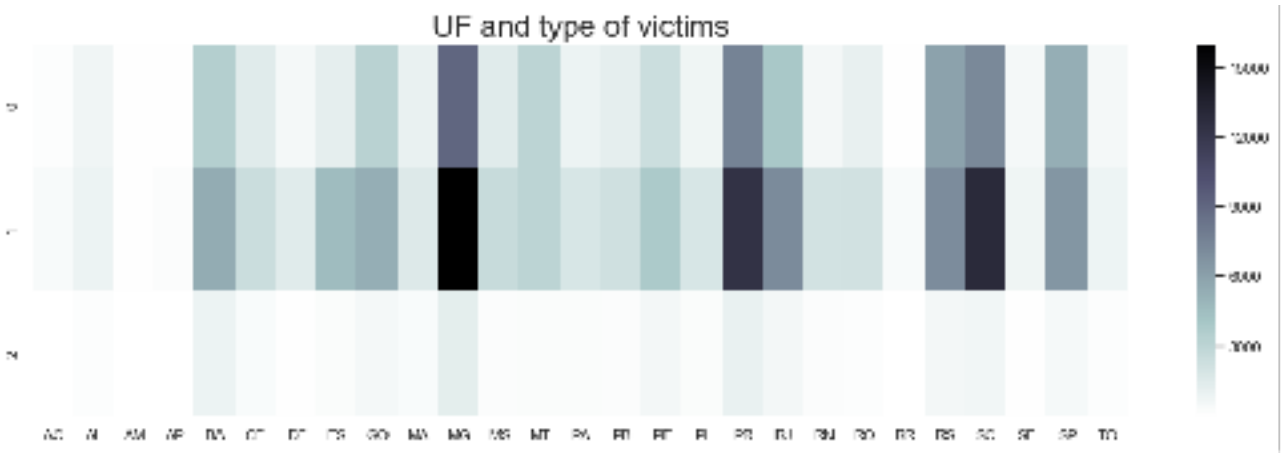
While most of the accidents with injured victims occurred during day time (61 thousand), accidents during the night with dead victims were higher than during the day, probably due to worse visibility causing accidents with more fatal victims.



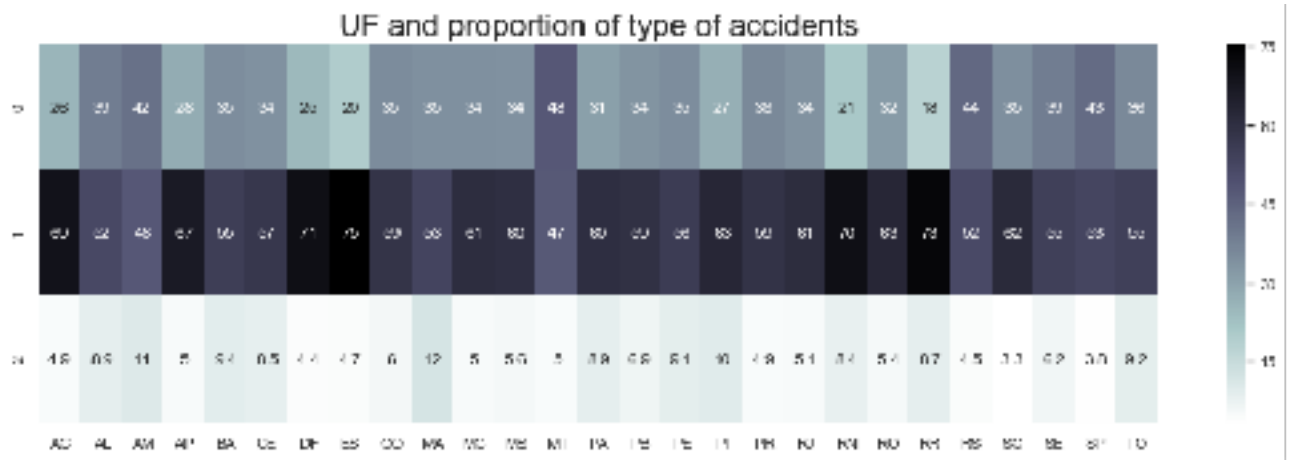

It is possible to see that some causes of accidents are highly correlated to the type of accident caused. For instance, rear collision are highly associated with unsafely distance and incompatible speed can be associated with the vehicle being out of the road.



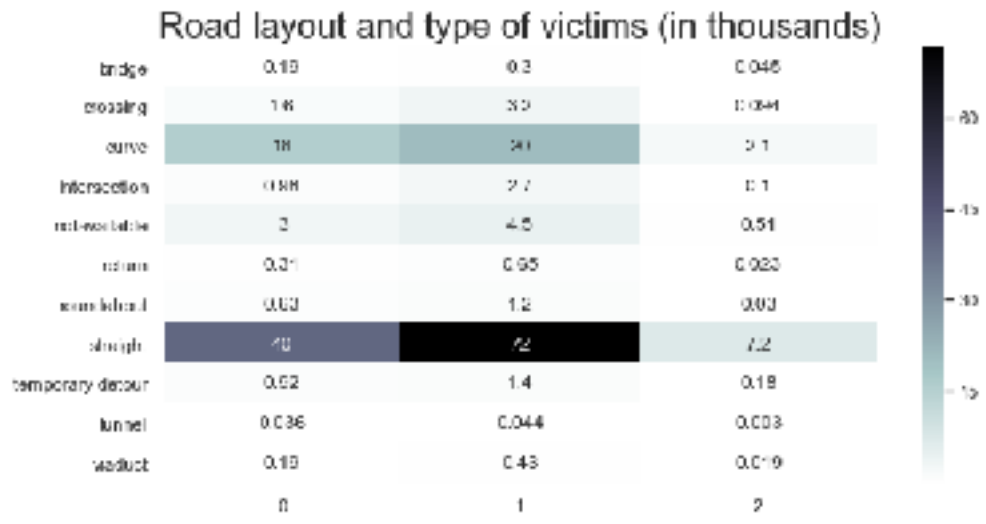

We can see that the state of MG, PR and SC are the states that have more accidents, followed by SP, RS and RJ. Although having a small number of accidents (2961), MA is the state with the highest number of accidents with dead victims in proportion (12.2%). SC, for instance has 10 times the number of accidents than MA but only 3.3% of them had dead victims.



AM, MA and PI are the states with more than 10% of accidents resulting in dead victims while AC, DF, ES, PR, RS, SC and SP have less than 5% of accidents resulting in dead victims. Almost 50% of accidents in MT have no victims, which makes it the most secure state, considering this prerogative.

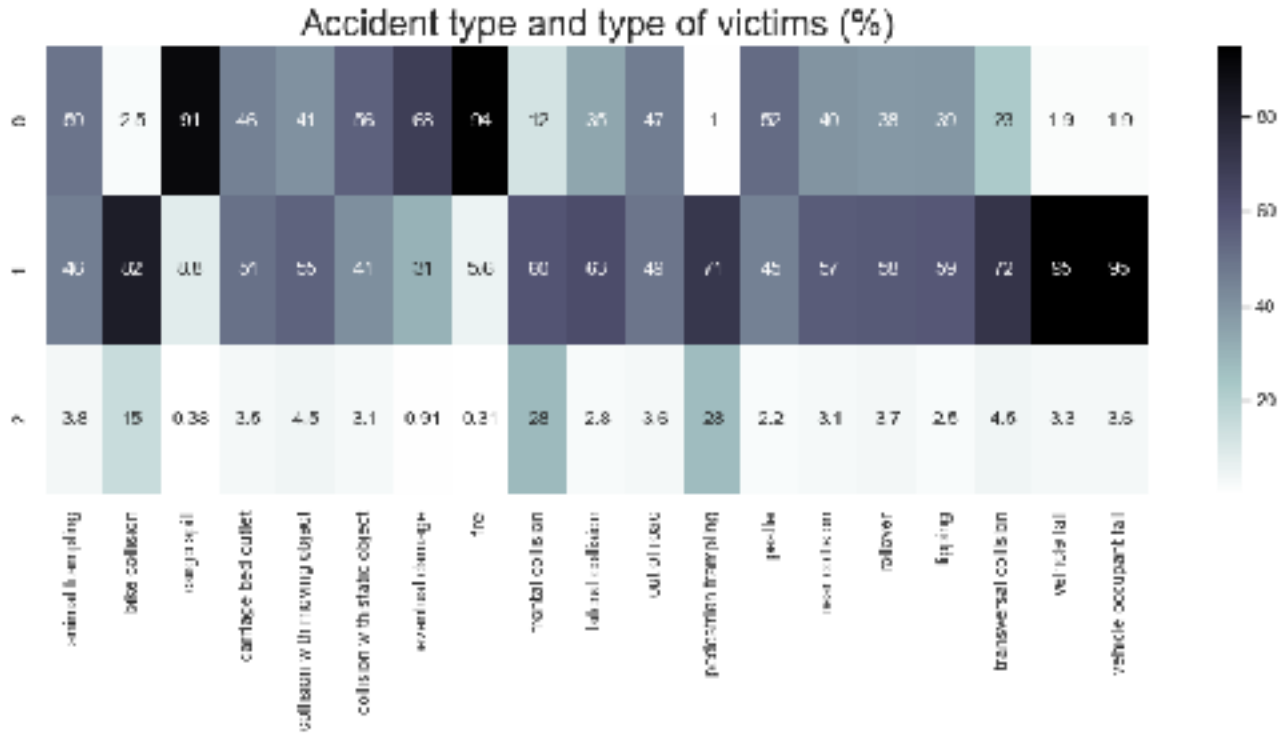



Most accidents occur in straight highways followed by accidents in curves. In general, one every 10 accidents with victims have dead victims.




Accidents in bridges have a higher chance to have dead victims (8,5%), followed by accidents in temporary detours (7,5%) while accidents in roundabout have the lowest chance of dead victims (1.6%). Be the other hand, accidents in tunnel have the higher percentage of accidents with no victims (43%) followed by accidents in curves (41%).


It is possible to observe in the next figure, that accidents involving fire and cargo spill has a high chance of having no victims (> 90%). Accidents with vehicle fall, vehicle occupant fall and bike collision have a high chance of having injured victims (> 80%). Frontal collision and pedestrian trampling have a huge chance of having dead victims (28%) while bike collision has a 15% chance of having dead victims. Thus, we can say that pedestrian trampling, frontal collision and bike collision are the most dangerous type of accidents.




## Algorithms and Techniques

All main classifier from the *sklearn* library were used in order to identify the best one for the present study. The idea was to start with simpler models, such as Logistic Regression, go further deep with ensemble models (Random Forest and Gradient Boosting) up to more

complex models, such as Neural Networks. The following models were used on the present study:

- Random Forest Classifier
- Gaussian Naive Bayes
- Logistic Regression
- Support Vector Machines (SVM)
- KMeans
- Gradient Boosting
- Neural Networks

The models Logistic Regression, Random Forest, Gradient Boosting and SVM were applied using the GridSearchCV method for optimising the hyper-parameters. The other models were used in their default configuration.

The techniques used were:

- Configuring hyper-parameters using GridSearchCV
- Under-sampling: selecting the same number of records for each class to avoid unbalanced classes
- Reducing classes: instead of trying to estimate 3 classes of victims, try to predict whether there would be a victim or not (binary output)
- Change the output from a categorical variable (type of injured) to a numerical parameter (number of victims)
- Changes in the input variables
- Use three simple models to predict each one of the three main classes (binary outputs) and use the probability results as features to a more complex model to predict the three classes

### Benchmark

Chong, Abraham & Paprzycki, 2005 did a similar study, where they used neural network for trying to classify victims degree of injuries on traffic accidents. On this study, they had 5 classes of injures, including no injury, possible injury, non-incapacitating injury, incapacitating injury, and fatal injury. The features where based both on road and driver's characteristics. The road characteristics were similar to what we found in the datasets of National Road Police. The authors found a ~60% accuracy on predicting the type of victims.

# III. Methodology

## Data Preprocessing

The fixes made on the dataset were:

- Fix the date to the pattern yyyy-mm-dd
- Removed values that did not have the target variable
- Add columns for hour, year, month and day of the accident
- Change the name of the target variable to classes 0, 1 and 2
- Renamed and standardized weekdays to English
- Renamed and standardized accident type to English
- Added a simplified accident type column
- Renamed and standardized accident cause to English
- Renamed and standardized accident type to English

- Renamed and standardized climate to English
- Renamed and standardized moment of day to English
- Rounded the km variable
- Define the br variable as object
- Renamed and standardized road type to English
- Renamed and standardized road layout to English
- Renamed and standardized road way to English
- Removed datasets were climate and moment of day were not available

## Implementation

The first step was to construct a quick and dirt model (notebook 01) using Logistic Regression. An accuracy of 0.63 was achieved. However, the recall of class 2 (dead victims) was only 0.01, which is not nearly good enough. After that, all other models were compared with this baseline in an attempt to achieve the best model.

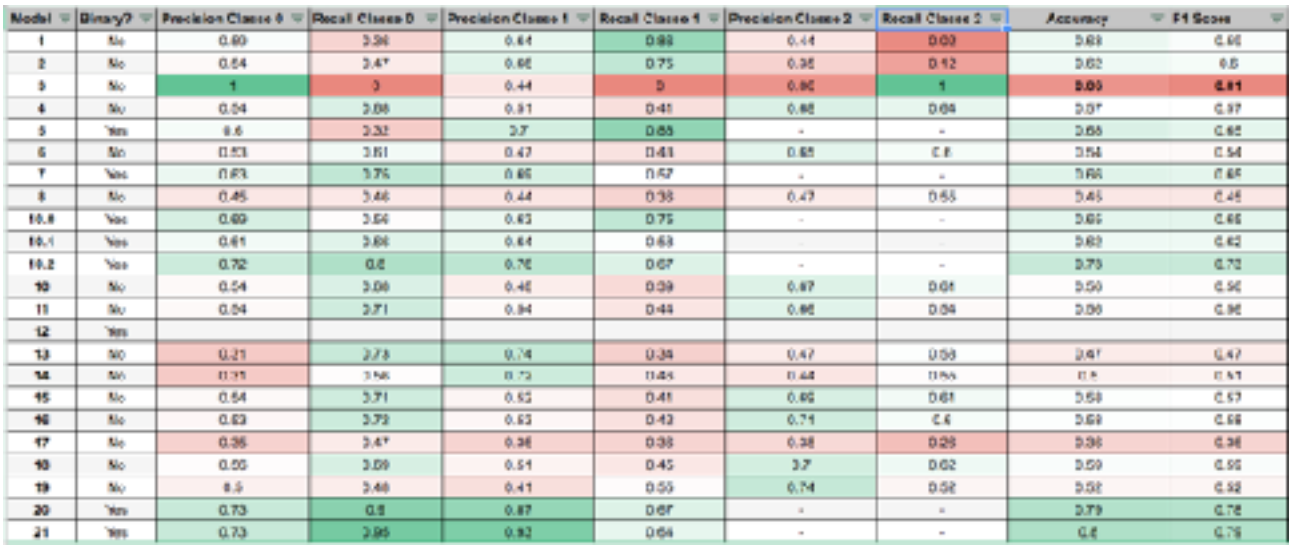
Implementation table:

| ID | Num. classes | Model | Techniques | Num of records considered | Features |
|---|---|---|---|---|---|
| **1** | 3 | Logistic Regression | GridSearchCV | 180991 | weekday, uf, br, km, accident cause, accident type , moment_of_day climate, road_layout |
| **2** | 3 | Random Forest | GridSearchCV | 180991 | weekday, uf, br, km, accident cause, accident type , moment_of_day climate, road_layout |
| **3** | 3 | Gaussian Naive Bayes | None | 88038 | weekday, uf, br, km, accident cause, accident type , moment_of_day climate, road_layout |
| **4** | 3 | Logistic Regression | GridSearchCV and undersampling | 31365 | weekday, uf, br, km, accident cause, accident type , moment_of_day climate, road_layout |
| **5** | 2 | Logistic Regression | GridSearchCV and binary output | 180991 | weekday, uf, br, km, accident cause, accident type , moment_of_day climate, road_layout |
| **6** | 3 | Random Forest | GridSearchCV and undersampling | 31611 | weekday, uf, br, km, accident cause, accident type , moment_of_day climate, road_layout |
| **7** | 3 | Logistic Regression | GridSearchCV, binary output, under sampling | 129333 | weekday, uf, br, km, accident cause, accident type , moment_of_day climate, road_layout |

| ID | Num. classes | Model | Techniques | Num of records considered | Features |
|---|---|---|---|---|---|
| **8** | 3 | Logistic Regression | GridSearchCV and undersampling | 31365 | weekday, uf, br, km, moment_of_day, climate, road_layout |
| **9** | 0 | Random Forest | GridSearchCV and numerical output | 88038 | weekday, uf, br, km, accident cause, accident type , moment_of_day climate, road_layout |
| **10** | 3 | Logistic Regression | GridSearchCV, model composition and undersampling | 88038 | weekday, uf, br, km, accident cause, accident type , moment_of_day climate, road_layout |
| **11** | 3 | Logistic Regression | GridSearchCV and unbalanced undersampling | 42291 | weekday, uf, br, km, accident cause, accident type , moment_of_day climate, road_layout |
| **12** | 2 | Random Forest | GridSearchCV, binary output and undersampling | 128332 | weekday, uf, br, km, accident cause, accident type , moment_of_day climate, road_layout |
| **13** | 3 | Gaussian Naive Bayes | Unbalanced undersampling | 51455 | weekday, uf, br, km, accident cause, accident type , moment_of_day climate, road_layout |
| **14** | 3 | SVM | GridSearchCV and undersampling | 30873 | weekday, uf, br, km, accident cause, accident type , moment_of_day climate, road_layout |
| **15** | 3 | Logistic Regression | GridSearchCV and undersampling | 30873 | weekday, uf, br, km, accident cause, accident type , moment_of_day climate, road_layout, month, hour, day, year |
| **16** | 3 | Random Forest | GridSearchCV and undersampling | 30873 | weekday, uf, km, accident type , climate, month, hour, day, year |
| **17** | 3 | KMeans | Undersampling | 30873 | weekday, uf, br, km, accident cause, accident type , moment_of_day climate, road_layout, month, hour, day, year |
| **18** | 3 | Gradient Boosting | GridSearchCV and undersampling | 30873 | weekday, uf, km, accident type , climate, month, hour, day, year |
| **19** | 3 | Neural Network | Undersampling | 30873 | weekday, uf, km, accident type , climate, month, hour, day, year |

| ID | Num. classes | Model | Techniques | Num of records considered | Features |
|---|---|---|---|---|---|
| **20** | 2 | Neural Network | Unbalanced undersampling, binary output | 41166 | weekday, uf, br, km, accident cause, accident type, climate, road_layout, month, hour, day, year |
| **21** | 2 | Gradient Boosting | GridSearchCV, binary output and undersampling | 41166 | weekday, uf, br, km, accident cause, accident type, climate, road_layout, month, hour, day, year |

For each implementation described above, the result of precision and recall and the general accuracy and F1 score were collected to define which model was best. The table below sums up the results found:



| Model | Binary? | Precision Classe 0 | Recall Classe 0 | Precision Classe 1 | Recall Classe 1 | Precision Classe 2 | Recall Classe 2 | Accuracy | F1 Score |
|---|---|---|---|---|---|---|---|---|---|
| 1 | No | 0.60 | 0.36 | 0.64 | 0.83 | 0.44 | 0.03 | 0.63 | 0.56 |
| 2 | No | 0.64 | 0.47 | 0.66 | 0.75 | 0.36 | 0.12 | 0.62 | 0.6 |
| 3 | No | 1 | 0 | 0.44 | 0 | 0.66 | 1 | 0.69 | 0.61 |
| 4 | No | 0.54 | 0.06 | 0.81 | 0.41 | 0.66 | 0.04 | 0.57 | 0.37 |
| 5 | Yes | 0.6 | 0.32 | 0.7 | 0.85 | - | - | 0.66 | 0.65 |
| 6 | No | 0.53 | 0.61 | 0.47 | 0.43 | 0.65 | 0.8 | 0.54 | 0.54 |
| 7 | Yes | 0.63 | 0.75 | 0.65 | 0.57 | - | - | 0.66 | 0.65 |
| 8 | No | 0.45 | 0.46 | 0.44 | 0.38 | 0.47 | 0.55 | 0.45 | 0.45 |
| 10.0 | Yes | 0.69 | 0.64 | 0.63 | 0.75 | - | - | 0.65 | 0.66 |
| 10.1 | Yes | 0.61 | 0.66 | 0.64 | 0.53 | | | 0.63 | 0.62 |
| 10.2 | Yes | 0.72 | 0.6 | 0.76 | 0.67 | - | - | 0.75 | 0.72 |
| 10 | No | 0.54 | 0.06 | 0.45 | 0.09 | 0.67 | 0.01 | 0.59 | 0.56 |
| 11 | No | 0.54 | 0.71 | 0.64 | 0.44 | 0.66 | 0.04 | 0.59 | 0.56 |
| 12 | Yes | | | | | | | | |
| 13 | No | 0.21 | 0.73 | 0.74 | 0.39 | 0.47 | 0.03 | 0.47 | 0.47 |
| 14 | No | 0.21 | 0.56 | 0.73 | 0.45 | 0.44 | 0.05 | 0.5 | 0.51 |
| 15 | No | 0.54 | 0.71 | 0.63 | 0.41 | 0.66 | 0.01 | 0.59 | 0.57 |
| 16 | No | 0.63 | 0.73 | 0.63 | 0.43 | 0.71 | 0.6 | 0.69 | 0.68 |
| 17 | No | 0.36 | 0.47 | 0.36 | 0.38 | 0.38 | 0.28 | 0.36 | 0.36 |
| 18 | No | 0.55 | 0.09 | 0.51 | 0.45 | 0.7 | 0.02 | 0.59 | 0.55 |
| 19 | No | 0.5 | 0.46 | 0.41 | 0.55 | 0.74 | 0.02 | 0.52 | 0.52 |
| 20 | Yes | 0.73 | 0.6 | 0.87 | 0.87 | - | - | 0.79 | 0.78 |
| 21 | Yes | 0.73 | 0.86 | 0.92 | 0.64 | - | - | 0.8 | 0.78 |

The model 9, which tried to compute the number of people injured instead of the type of victims, was not added to this table, since it is not a classification approach. Nevertheless, this model had a perfomance not much better than predicting the mean number of injured people and was discarded as an inadequate model.

We can observe that the model 10, which is a combination of 3 models into a fourth model, did not have a better perfomance than the other more simple models.

Undersampling was a technique much more efficient than changing algorithms or trying more or less variables. Some variables, such as the number of the road or the accident cause, initially thought as important features, actually weren't as important. We ended up with 2 models: a model with a binary output (with out without victims) with an accuracy of 0.8 and a model with a multi class output, with accuracy of 0.59.

### Refinement

The solution was not ideal (over 0.6 of accuracy), as define in the benchmark session, for the three class model, but close enough to define de model as a good enough. However, the model with a binary class got an accuracy of 0.8, which is very good.

We could try to improve the multi-class model with adding more records, since the under sampling made a huge decrease in the number of records processed.

# IV. Results

### Model Evaluation and Validation

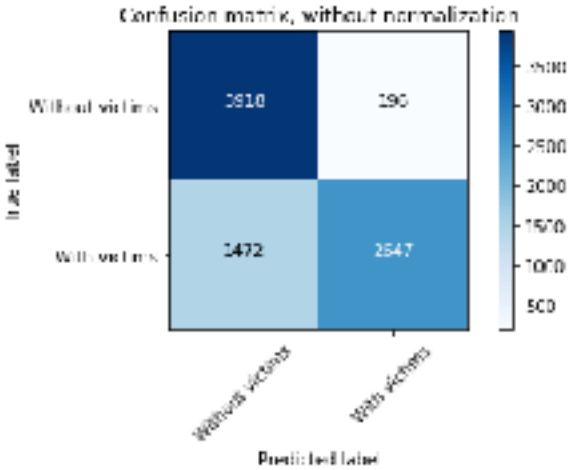
The model was considered similar, in therms of accuracy, of the model proposed as benchmark in the begging of the study. Even though the original proposition (multiclass model) had a similar accuracy to the benchmark model, a binary model had a very good accuracy, that can be considered good.

The database was divided between train and test samples, so we could test the model with a dataset that would not be known to it previously.

The results here presented are a combination of the best algorithm and variables, but this were small changes compared to the change that could be achieved only by under sampling and balancing classes through this technique.

### Binary Model

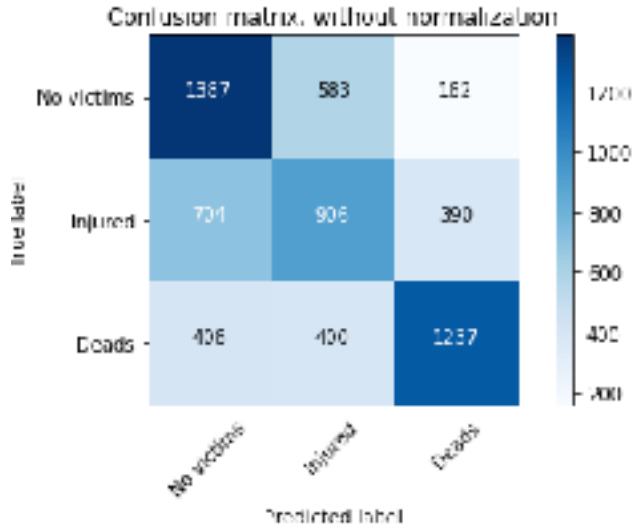
On the figure below we can see the confusion matrix of the best binary model (notebook 21). We can see that only 196 accidents that had not victims were classified as with victims, while 1472 with victims were classified as without victims. The table below has the results of precision, recall and f1-score for each class.




| | | | | support |
|---|---|---|---|---|
| 0 | 0.73 | 0.95 | 0.82 | 4114 |
| 1 | 0.93 | 0.64 | 0.76 | 4119 |
| avg / total | 0.83 | 0.80 | 0.79 | 8233 |

### Multi class Model

On the confusion matrix below, we can observe that although class 0 (no victims) and class 2 (dead victims) had high accuracy, the class 1 (injured victims) had a worse performance. This can be seen on the table below resulting in both precision an recall factors being below 0.5. Several approaches were made trying to make this perfomance improve, but none of them succeeded.




| | precision | recall | f1-score | support |
|---|---|---|---|---|
| 0 | 0.56 | 0.65 | 0.60 | 2132 |
| 1 | 0.48 | 0.45 | 0.47 | 2000 |
| 2 | 0.69 | 0.61 | 0.65 | 2043 |
| avg / total | 0.58 | 0.57 | 0.57 | 6175 |

# V. Conclusion

### Reflection

I believe that the models had a good result, similar to those found on the study define as the benchmark. This study could improve defining priority of ambulance attendance, specially in places where the recurses are limited. I would like to try new approaches to increase the accuracy of them. Some approaches that could be viable are: increase number of records, test different algorithms and try to gather more information about the accidents.

It was really difficult working with multiclass models, since it is more intuitive the possible approaches that could be done. I believe that the binary model achieved a good performance that could help places were recurses are scarce, but the recall of the class of injured should be increased to prevent putting people in risk with such a model.