# Road Traffic Injuries Classifier

## Abstract

This repository is dedicated to predicting road traffic injuries. The used data is collected from Kaggle of about 131956 entries split into different files, and over 25 features. The overall goal here is to predict 'injury severity', which is a multi-class classification issue, based on analyzing the provided dataset. According to data specifications, injured people are classified into four groups of 1. Unharmed 2. Killed 3. Injured hospitalized 4. Slightly injured. The data was collected in 2019 in France. Thus, for clarity purposes, we had to translate the columns into English.

## Data

The first data file holds the accident characteristics data. It has 58413 entries and 15 columns (possible features). The columns are of different types; int, float, and objects. Similarly, the second data file holds the users' (drivers) data with 132977 entries and 15 columns. The users involved in an accident include car drivers, bike riders, pedestrians, and their companies.

## Model Design

- Data Cleaning:

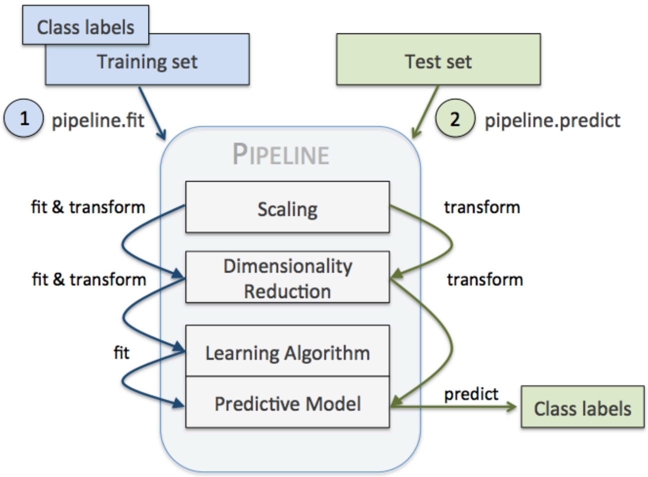
Removing null and duplicate values, translating columns into English, and merging the data into a single data frame for further processing.

- Feature engineering:

After cleaning the data, we ran a correlation analysis on the included feature. We found that `user_place` is highly correlated with and `user_category`(0.9) and `ped_company`(0.86). Thus, we eliminated the latter two and kept `user_place`.

We also eliminate other features (column) such as the `year`, as the whole dataset is dedicated for 2019 traffic injuries. Eliminated columns also include the address and `municipality` as they can be substituted by other existing ones such as `longitude`, `latitude`, and `department`.

After Data Cleaning and Features Engineering, the plan is to follow the steps depicted in the next picture.



- Algorithms :
  - Oversampling: Random, SMOTE
  - Classification:

    (1)Logistic Regressing,<br>
    (2)KNN,<br>
    (3)Decision Tree,<br>
    (4)Random Forest Tree, and<br>
    (5)Support vector machines

**Tools**

* Data manipulation and analysis: Numpy and Pandas for numeric data
* Data modeling: Scikit-learn
* Plotting and Visualization: Matplotlib and Seaborn