GRANDE ÉCOLE D'ACTUARIAT & GESTION DES RISQUES

The car accidents in France from 2005 to 2016 and their severity

ABBOU Keren, CANETE Sarah, HOUSSEIN KASSIM Saïda

Thanks

First of all, we would like to thank the board of examiners for having allowed us to lead this project.

We would like to thank Mr. CLOT, our referent teacher, for his constant help, advice and supervision.

Also, we would like to thank Mrs. HAVET, who helped us a lot for the econometric part and whose lessons were so helpful and interesting.

Thanks to Mr. GOFFARD, who helped us to begin our project.

We also are very grateful to ISFA and its Econometric and Statistics Master, its teachers and their lessons. They provided us a real intellectual satisfaction and helped us develop our reflexion.

We thank Mr. ALMENDRA too, who taught us how to prioritize and how to organize our project.

Contents

Chapter 1

Introduction

According to the "Securite routiere" website[1], related to the French Road Safety observatory, some studies showed that there were 3,477 people killed in an accident and 72,645 injured people among which 27,187 were hospitalized in France, in 2016.

Because there are still many road accidents nowadays, we decided to study the ones that occurred in France from 2005 to 2016. We will study this topic and also analyze which factors do have a real impact onto the severity of the accidents. It was important for us to deepen this subject and the reasons why an accident is more serious than another one, as we thought that by showing how many accidents occur and by determining the factors that have an impact onto their severity, we could help people to become aware of the dangers of the road, and give some advice to decrease the severity of an accident, when one cannot prevent it.

We hope that we will be able to share them with as many people as possible, firstly while making our presentation, and then by making our report available at ISFA.

Our data source was KAGGLE.com, as suggested by our referent teacher, Mr. CLOT. It initially contained 4 different datasets, all about car accidents in France from 2005 to 2016 and all linked by an arbitrary accident id. We had a dataset Vehicles, containing information about the implicated vehicles like their types, a dataset users about the injured people (their ages, injuries etc.), a dataset characteristics containing many parameters like the presence or absence of a security system, and a dataset places about the place of the accident (in or out an agglomeration, etc.).

First of all, we will consider the quality of our data : we have many missing values and outliers. We will explain what a missing value and an outlier are, then we will discuss the different methods to treat them, such as the K-nearest-neighbors method, which is precisely the one we chose. Next, we will explain the way we applied it, and we will then check if our results are relevant enough.

In a second part, we will make some data visualization, using our initial and inputed values, in order to show graphically what our data can teach us about the accidents in France from 2005 to 2016.

[1] `https://www.preventionroutiere.asso.fr/2016/04/22/statistiques-daccidents/`

In a last part, we will analyze the severity of car accidents, by making an econometric study. We will first explain what an econometric study is used for, and how it theoretically works in our case. After that, we will justify the new data treatments we made on our table and then, we will determine the factors that have a real impact onto the severity of these accidents. Finally, we will present and interpret our results.

In conclusion, we will give some advice, according to our results, so that people can reduce the risk of having an accident and then, we will make some others, in order to raise people's awarness of the factors they may control to minimize the severity of accidents when they cannot prevent them.

Chapter 2

Data mining

2.1 First manipulations of the data

As explained in the introduction, at first, we had 4 different tables, all containing different variables about the accidents in France from 2005 to 2016. They were linked by an arbitrary accident id.

We used SAS to merge them, and so we obtained a large dataset containing all the variables from all of the 4 initial tables. We could therefore start to analyze our data on an unique dataset. This was necessary, as STATA can only load one dataset at a time, and we needed to study variables from all of these datasets for the econometric regression.

As we did not know the meaning of every variable (they were not mentioned anywhere on KAGGLE.com and it was impossible for us to guess them), we decided to delete these unknown variables and to work on the others.

Then, we analyzed the variables and determined for each what the missing values were, because some already were identified as "NA", but others were as ".".

We also found out that some variables were more complicated : for example, the "Secu" variable was supposed to be composed of two numbers. The first one, from 1 to 9, corresponded to the type of the security system, and the second one, between 1 and 3, corresponded to the presence, the absence or the undetermined use of it.

Some values only contained only one number, others ending by 3, others containing numbers we did not know the meaning etc. That's why we considered that all these values were missing.

After having determined all the missing values of every variable, we identified all of them under the same form : "NA", so that the software could detect them correctly.

Thus, we could start our project.

2.2 The missing values and outliers : theoretical part

2.2.1 Problematic

The first issue we came across when we got the table was the presence of missing values. Even if the amount of available data is increasing and if there is an emergence of Big Data, the missing data issue remains widespread in statistical problems and requires a specific approach.

It is necessary to treat the missing values in order to fill the information, without altering the meaning of the initial dataset.

The analysis of the missing data is an essential point in the data analysis. It is important to know which errors lead to analysis mistakes in their consideration.

The context is very important when dealing with the meaning of the missing values.
Before the data can be used, the input dataset must be completed. Otherwise, we may not be able to interpret and analyze it.

Ignoring missing data can result in significant bias in analysis models, as well as a loss of precision.

Moreover, modeling data-out behavior requires a deep understanding of the business sense of the manipulated data .
Missing data processing is a necessary phase for any Data Science project.

We can distinguish two types of missing data: intentional and unintentional ones.

- Intentional missing data: they are provided by the investigator, they may occur because the flow of the survey questions isn't good (for example: question 1: yes or no, if yes question 2, if no question 3), or because of the exclusion of some units from the sample.
- Unintentional missing data: they are not under the control of the investigator and may have multiple causes. For example, the subject skipped a question when he/she was answering the questionnaire, quit the study before the end of the follow - up, or was sampled but then refused to participate in the survey study. There may also be some problems in transmitting or entering data.

This explains why some information may be missing.

2.2.2 What is a missing value?

In a database, each line is an observation, and each column represents an individual characteristic (age, sex etc.).

In order to properly address the imputation of missing data, we need to understand the root causes, especially if they are not the result of chance. Missing data are common and could have a significant effect on the conclusions that can be drawn from the data.

The difficulties in dealing with missing data are the assumptions we make about lack of data patterns. They can be the result of non-response (during the survey), of various experimental problems (in biology), or of a bad data entry, of aberrations etc.

There are 2 types of missing data: they may be partial, that is, for a given individual, that only a few values are missing. These are called a feature. There are also some missing data called total, where all the variables for a given individual are unobserved.
The goal when dealing with missing data is to reconstruct a realistic vision of data.

It is always important to observe and understand the dataset before deciding which approach we have to use to handle missing data.

2.2.3 What is an outlier ?

An outlier is defined as an observation that deviates too much from the other observations, so it arouses suspicions that it was generated by a different mechanism than the other observations.

Inlier, on the other hand, is defined as an observation that is explained by underlying probability density function.

In clustering, outliers are considered as noise observations that should be removed in order to make more reliable clustering.

In data mining, detection of anomalous patterns in data is more interesting than detecting inlier clusters.

The exact definition of an outlier depends on the context.
Definitions fall roughly into five categories :

1. distribution-based
2. depth-based
3. distance-based
4. clustering-based
5. density-based.

Distribution-based methods originate from statistics, where the observation is considered as an outlier if it deviates too much from underlying distribution.

2.2.4 How can the missing values be distributed? Mechanism of missing values

It is very important to know the type of missing data in order to avoid errors leading to analysis bias in their consideration.

Also, to decide on the best approach to take the missing values into account in the analysis, it is necessary to do an exploratory analysis allowing the understanding the distribution of the missing values in the dataset.

Little and Rubin defined 3 types of missing data distributions :

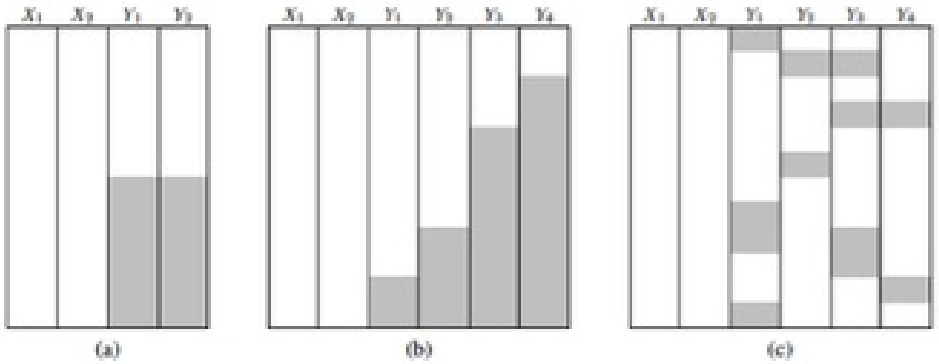


Figure 2.1: The 3 types of missing data distributions

It could be univariate , monotonous or without structure.

The missing data which are completely randomly distributed are called missing complete at random (MCAR). The probability of absence is the same for all observations. This probability depends only on external parameters, and is independent of the variables.

Missingness at random (MAR) happens when data is not missing in a completely randomly way. It's the case when the probability of absence is related to one or more other variables observed.

Missing data not randomly (MNAR) induces a loss of precision (inherent to any case of missing data) but also a bias that requires the use of sensitivity analysis.
When the probability of absence of a variable depends on the variable itself or other unobserved variables, the treatment is more complex.

2.2.5 The different types of imputation

There are different missing data imputation methods such as LOCF, median imputation, mean[1], or by modeling or statistical learning: regression and local regression, KNN (K-Nearest Neighbors), PLS, SVD, Random Forest regression or also by multiple imputation.

The use of the data-deletion method adds considerable bias, and creates a loss of power and rapidly significant cases. It is willing to use only available and complete observations.
Despite these disadvantages, it remains one of the most common strategies because it is usually imposed by some softwares and has the advantage that it doesn't require to use the specification of an imputation model (when the data is replaced, the quality conditions the results of the analysis).
If the data is not MCAR, removing observations will induce bias in the analysis since the subsample of the cases represented by the missing data may not be representative of the original sample.

The imputation method has advantages and disadvantages.
Despite the fact that its empirical efficiency has been shown in regression and classification trees, it is mainly valid in MAR and when covariates are strongly correlated.
Moreover, it is a rather time wisean inefficient method.
Missing data imputation consists in replacing missing values in the dataset with artificial values. Ideally, these replaced values should not lead to a significant alteration of the distribution and composition of the dataset, as they will play the same role subsequently as the observed values. There are several types of possible stationary completions: last observation carried forward (LOCF), or linear combination of observations (imputation by the mean or the median for example).

Local regression (LOESS) is also used to impute missing data.
For this, a polynomial of low degree is adjusted around the weighted least squares missing data, giving more importance to values that are close to the missing data.
There are other methods such as the NIPALS algorithm, the singular value decomposition, the use of random drill bits, Bayesian inference, multiple imputation or amelia 2.

For imputation methods, multiple imputation and KNN are preferable when the missing data proportion is large.

2.2.6 The KNN method

Completion by K-Nearest Neighbors (KNN) consists in executing the following algorithm that models and predicts missing data.
Another approach to simple imputation is to use the observed values of individuals that are similar to the individual for whom a value is missing.

The KNN method is a multivariate imputation method, based on a notion of distance between individuals, $d(i, i')$, obtained from q fully covariates, X.

[1] a code is proposed in the annexes part for mean and median imputations

For a missing value y_{ij}, the approach consists in calculating the set of distances $d(i, i')$, $\forall i \neq i'$, and keeping the K observations corresponding to the smaller distances.
It's possible to impute the missing value.

This method requires 2 parameters to be chosen: the distance and the number of neighbors for the estimation.

When the dataset contains categorical variables, we use a distance that takes into account the existence of these variables.
The closest neighbors are selected in an optimal correlation space with the variables to be imputed.
Regarding the K, it is necessary to choose a number depending on the size of our sample.

Thanks to the VIM package on R, we can make an imputation by KNN for mixed data.
To do this, the K neighbors are chosen using a variation of the Gower distance. This distance can be applied to a set of variables at once numeric, categorical and binary.
It is based on a notion of contribution of the covariate X_j which is defined by :

X_j is numerical

$$S_{ii'j} = \frac{\mid x_{ij} - x_{i'j} \mid}{max_i(x_{ij} - min_i(x_{ij}}$$

X_j is categorical

$$S_{ii'j} = \begin{cases} 1 & \text{si } x_{ij} = x_{i'j} \\ 0 & \text{sinon.} \end{cases}$$

Then, we can deduce a distance between individuals i and i' as follows :

$$d(i, i') = \frac{\sum_{j=1}^{p}(S_{ii'j})}{n}$$

The numeric variables are finally imputed by the median of the values of the neighbors whereas the categorical variables are imputed by the mode of the values of the neighbors.

The Knn method is not the most efficient but given the time, the data and the material we had, it was in our opinion, the most adequate to use.

Whatever the technique of processing missing data used, it will not replace a complete and consistent data set. Fixing missing data helps to "limit the damage" while preserving the composition of the dataset. However, no matter how much care and technique we use to process the missing data, there will always be a loss of information in our dataset and we may even insert involuntary biases that do not necessarily reflect relations governing our data.

Since the imputed values are estimated, it is important to check if they are coherent. In order to do this, we can use diagnostic tools. This usually consists in comparing imputed values with observed values either with the help of graphs or with the help of basic statistics.

Thanks to the notion of distances in performing a minimization to impute the missing value, the outliers are automatically detected and excluded.

2.2.7 Imputation check

The first approach to assess the quality of an imputation method is to over-impute, by removing observed data and comparing the imputed values to the actual values before deletion. This approach is relatively interesting for the evaluation of the quality of a given method.

An alternative approach is to use only observed values and their distribution to evaluate the relevance of the imputed values.

The use of general diagnostic tools helps to check if the imputation is good or not.

The imputation error is often limited to the comparison between observed and imputed values. There are several types of imputation errors:

- the measurement error: it is the error made between the observed values and the true values for the individuals (it remains unknown because of errors related to the measuring tool or for instance, uncontrolled experimental differences between the measurements).
- the pure error: it is the imputation model error that is specified in the context of an imputation approach, in which the variable with missing values is imputed from a model involving only completely observed covariate.

2.2.8 Going further

In order to measure the impact of the imputation and to quantify its errors, the most common approach is to repeat the imputation several times by introducing hazard. These approaches are known as multiple imputation.

Multiple imputation consists in proposing, for each missing value, not one but several plausible values for imputation. This method is used to measure the variability, on the final result, of the imputation process.

Multiple imputation takes place in three phases:

- Imputation phase: the initial data table is duplicated M times and an imputation model is applied to each new data table. A part of random is introduced, the duplication of the original table (which is not identically reproduced) or the imputation itself allow to obtain M different tables with completed data.
- Statistical analysis phase: the chosen statistical analysis (regression, PCR, network inference, etc.) to analyze the data table is implemented on each table of imputed data, to obtain M estimates.

- Combined analysis phase: the results are combined according to the rules defined by Rubin, to obtain a single final estimation, or to estimate the variability of the results by a target statistical analysis performed on the completed data.

Imputation procedures are incorporating appropriate variability across M imputed dataset in the model.
These imputation methods reflect correctly the variability of the imputed data method, taking into account the intra-imputation variability (corresponding to the variability due to the method itself and noise in the data) but also the variability between imputations (attributable to the missing data).

We wish we could have tried to make a multiple imputation, but due to lack of time and lack of computer resources, it could unfortunately not be performed .

Also, we could have used machine learning.
One of the KNN methods can be found in the Class package on R. To do this, we give the value of our K, a train table and a test table. The goal is to create a table consisting of complete individuals (without missing values) and another one with incomplete individuals.
KNN classification for test set from training set. For each row of the test set, the closest (in Euclidean distance) training set vectors are found, and the classification is decided by majority vote, with links broken at random. If there are some for the nearest family, all candidates are included in the vote. Before applying this method in our table with missing values, we try on a table with results which are know to asses its efficiency, that is to say, to calculate the proportion of good classifications. Then, we could have chosen the best K by making a loop that varies K, and calculates the percentage of success for each KNN. It is important to make sure that we have a KNN that predicts well, by taking into account how much time the computer will need to do it. A graph (a plot with all the accuracy percentages as a function of K) would have allowed us to visually select the best K to chose (depending on the maximum of the curve).

Unfortunately, due to lack of time and equipment, we could not test these methods either.

In order to remedy the lack of computer capacities, these methods could have been run on the different cores of the computer. The idea is to divide the tasks that need to be done on different hearts in order to save time in the execution of our program. The father's heart reads the information, and distributes it in such a way that the hearts of the children are always keen to execute part of the program.

2.2.9 Conclusion

Missing data is a problem commonly encountered in statistical analysis, whatever the field of study. The most appropriate method to take this into account depends on multiple parameters such as the typology of missing values, the type of mechanism that has led to their generation, their distribution in the dataset and the expectations of the user in terms of statistical analysis.

2.3 The missing values and outliers: Practical part

For our project, it was necessary to solve this missing values problem.

2.3.1 STEP 1: Reading the data

Firstly, we read the data to observe them.

```
> head(completavecvm)
  V1 id   num_acc num_veh place catu grav sexe trajet secu locp an_nais an mois jour hrmn lum agg inter atm col com
1  1  1 2.005e+11     A01     1    1    2    1      1   11    0    1976  5    1   12 1900   3   2     1   1   3  11
2  2  2 2.005e+11     B02     1    1    3    2      3   11    0    1968  5    1   12 1900   3   2     1   1   3  11
3  3  3 2.005e+11     B02     2    2    1    1     NA   11    0    1964  5    1   12 1900   3   2     1   1   3  11
4  4  6 2.005e+11     B02     3    2    1    2     NA   11    0    1991  5    1   12 1900   3   2     1   1   3  11
5  5  4 2.005e+11     B02     4    2    1    1     NA   31    0    2004  5    1   12 1900   3   2     1   1   3  11
6  6  5 2.005e+11     B02     5    2    1    1     NA   11    0    1998  5    1   12 1900   3   2     1   1   3  11
      lat    lon dep catr voie circ nbv prof plan lartpc larrout surf catv
1 5051500 294400 590    3   41    2   2    1    1      0      63    1    7
2 5051500 294400 590    3   41    2   2    1    1      0      63    1    7
3 5051500 294400 590    3   41    2   2    1    1      0      63    1    7
4 5051500 294400 590    3   41    2   2    1    1      0      63    1    7
5 5051500 294400 590    3   41    2   2    1    1      0      63    1    7
6 5051500 294400 590    3   41    2   2    1    1      0      63    1    7
```

Thanks to the nrow function, we could observe that our table consisted in 1,876,005 individuals.

2.3.2 STEP 2: Preparation and exploration of the data

Then, we made the data preparation and exploration.

The function str allowed us to see that there were 35 characteristics that were mostly integers.

```
> str(completavecvm)
'data.frame':   1876005 obs. of  35 variables:
 $ V1     : int  1 2 3 4 5 6 7 8 9 10 ...
 $ id     : int  1 2 3 6 4 5 7 8 9 10 ...
 $ num_acc: num  2.01e+11 2.01e+11 2.01e+11 2.01e+11 2.01e+11 ...
 $ num_veh: chr  "A01" "B02" "B02" "B02" ...
 $ place  : int  1 1 2 3 4 5 1 1 1 1 ...
 $ catu   : int  1 1 2 2 2 2 1 1 1 1 ...
 $ grav   : int  2 3 1 1 1 1 1 3 1 3 ...
 $ sexe   : int  1 2 1 2 1 1 1 1 1 1 ...
 $ trajet : int  1 3 NA NA NA NA 5 5 1 1 ...
 $ secu   : int  11 11 11 11 31 11 11 21 21 21 ...
 $ locp   : int  0 0 0 0 0 0 0 0 0 0 ...
 $ an_nais: int  1976 1968 1964 1991 2004 1998 1955 1979 1983 1956 ...
 $ an     : int  5 5 5 5 5 5 5 5 5 5 ...
 $ mois   : int  1 1 1 1 1 1 1 1 1 1 ...
 $ jour   : int  12 12 12 12 12 12 21 21 21 21 ...
 $ hrmn   : int  1900 1900 1900 1900 1900 1900 1600 1600 1845 1845 ...
 $ lum    : int  3 3 3 3 3 3 1 1 3 3 ...
 $ agg    : int  2 2 2 2 2 2 2 2 1 1 ...
 $ inter  : int  1 1 1 1 1 1 1 1 1 1 ...
 $ atm    : int  1 1 1 1 1 1 1 1 2 2 ...
 $ col    : int  3 3 3 3 3 3 1 1 1 1 ...
 $ com    : int  11 11 11 11 11 11 51 51 51 51 ...
 $ lat    : int  5051500 5051500 5051500 5051500 5051500 5051500 5053700 5053700 5054600 5054600 ...
 $ lon    : chr  "294400" "294400" "294400" "294400" ...
 $ dep    : int  590 590 590 590 590 590 590 590 590 590 ...
 $ catr   : int  3 3 3 3 3 3 2 2 2 2 ...
 $ voie   : chr  "41" "41" "41" "41" ...
 $ circ   : int  2 2 2 2 2 2 NA NA NA NA ...
 $ nbv    : int  2 2 2 2 2 2 2 2 0 0 ...
 $ prof   : int  1 1 1 1 1 1 1 1 1 1 ...
 $ plan   : int  1 1 1 1 1 1 1 1 1 1 ...
 $ lartpc : int  0 0 0 0 0 0 0 0 0 0 ...
 $ larrout: int  63 63 63 63 63 63 100 100 0 0 ...
 $ surf   : int  1 1 1 1 1 1 1 1 2 2 ...
 $ catv   : int  7 7 7 7 7 7 7 2 2 2 ...
```

To begin, we needed to understand our missing values. For this, we used a function that sums up the missing values of each column. We found out that there were too many missing values for our lat and lon variables.

```
> sapply(completavecvm,function(x) sum(is.na(x)))
     V1      id num_acc num_veh   place    catu    grav    sexe  trajet    secu    locp an_nais      an    mois    jour    hrmn
      0       0       0       0  100366       0       0       0  550885  409298    1664    2351       0       0       0       0
    lum     agg   inter     atm     col     com     lat     lon     dep    catr    voie    circ     nbv    prof    plan  lartpc
      0       0     253     116      24       6 1321219 1366644       0       2  124484   95494    4063  142248  124449   21272
larrout    surf    catv
  18297   58949      22
```

The following function allowed us to calculate the percentage of missing values for each variable.

```
> apply(completavecvm,2,pMiss)
         V1          id      num_acc      num_veh        place         catu         grav         sexe       trajet
0.000000e+00 0.000000e+00 0.000000e+00 0.000000e+00 5.349986e+00 0.000000e+00 0.000000e+00 0.000000e+00 2.936479e+01
       secu         locp      an_nais           an         mois         jour         hrmn          lum          agg
2.181753e+01 8.869912e-02 1.253195e-01 0.000000e+00 0.000000e+00 0.000000e+00 0.000000e+00 0.000000e+00 0.000000e+00
      inter          atm          col          com          lat          lon          dep         catr         voie
1.348610e-02 6.183352e-03 1.279314e-03 3.198286e-04 7.042726e+01 7.284863e+01 0.000000e+00 1.066095e-04 6.635590e+00
       circ          nbv         prof         plan       lartpc     larrout         surf         catv
5.090285e+00 2.165772e-01 7.582496e+00 6.633724e+00 1.133899e+00 9.753172e-01 3.142262e+00 1.172705e-03
```

Therefore, variables that had more than 50% missing values were removed from the rest of the analysis.

With the function md.pattern, we could see how the missing values were distributed. In the main body of the output table, “1” indicates a nonmissing value and “0” indicates a missing value. The first column shows the number of unique missing data patterns.

The rightmost column shows the number of missing variables in a particular missing pattern.

For example, the first row has no missing value and there are only “0” in the row.

The last row counts the number of missing values for each variable.

There are 236321 complete data.

```
> md.pattern(completavecvm)
       V1 id num_acc num_veh catu grav sexe an mois jour hrmn lum agg dep catr com catv col atm inter locp
236321  1  1       1       1    1    1    1  1     1    1    1   1   1   1    1   1    1   1   1     1    1
23759   1  1       1       1    1    1    1  1     1    1    1   1   1   1    1   1    1   1   1     1    1
42      1  1       1       1    1    1    1  1     1    1    1   1   1   1    1   1    1   1   1     1    1
569469  1  1       1       1    1    1    1  1     1    1    1   1   1   1    1   1    1   1   1     1    1
80777   1  1       1       1    1    1    1  1     1    1    1   1   1   1    1   1    1   1   1     1    1
7216    1  1       1       1    1    1    1  1     1    1    1   1   1   1    1   1    1   1   1     1    1
31      1  1       1       1    1    1    1  1     1    1    1   1   1   1    1   1    1   1   1     1    1
227981  1  1       1       1    1    1    1  1     1    1    1   1   1   1    1   1    1   1   1     1    1
24817   1  1       1       1    1    1    1  1     1    1    1   1   1   1    1   1    1   1   1     1    1
2450    1  1       1       1    1    1    1  1     1    1    1   1   1   1    1   1    1   1   1     1    1
125632  1  1       1       1    1    1    1  1     1    1    1   1   1   1    1   1    1   1   1     1    1
10700   1  1       1       1    1    1    1  1     1    1    1   1   1   1    1   1    1   1   1     1    1
1611    1  1       1       1    1    1    1  1     1    1    1   1   1   1    1   1    1   1   1     1    1
73525   1  1       1       1    1    1    1  1     1    1    1   1   1   1    1   1    1   1   1     1    1
16763   1  1       1       1    1    1    1  1     1    1    1   1   1   1    1   1    1   1   1     1    1
1258    1  1       1       1    1    1    1  1     1    1    1   1   1   1    1   1    1   1   1     1    1
2       1  1       1       1    1    1    1  1     1    1    1   1   1   1    1   1    1   1   1     1    1
29525   1  1       1       1    1    1    1  1     1    1    1   1   1   1    1   1    1   1   1     1    1
7586    1  1       1       1    1    1    1  1     1    1    1   1   1   1    1   1    1   1   1     1    1
550     1  1       1       1    1    1    1  1     1    1    1   1   1   1    1   1    1   1   1     1    1
12569   1  1       1       1    1    1    1  1     1    1    1   1   1   1    1   1    1   1   1     1    1
1507    1  1       1       1    1    1    1  1     1    1    1   1   1   1    1   1    1   1   1     1    1
70      1  1       1       1    1    1    1  1     1    1    1   1   1   1    1   1    1   1   1     1    1
3970    1  1       1       1    1    1    1  1     1    1    1   1   1   1    1   1    1   1   1     1    1
741     1  1       1       1    1    1    1  1     1    1    1   1   1   1    1   1    1   1   1     1    1
45      1  1       1       1    1    1    1  1     1    1    1   1   1   1    1   1    1   1   1     1    1
2554    1  1       1       1    1    1    1  1     1    1    1   1   1   1    1   1    1   1   1     1    1

       an_nais  nbv larrout lartpc  surf  circ  place  plan  voie  prof  secu trajet  lat  lon  tot
236321       1    1       1      1     1     1      1     1     1     1     1      1    1    1    0
23759        1    1       1      1     1     1      1     1     1     1     1      1    1    0    1
42           1    1       1      1     1     1      1     1     1     1     1      1    0    1    1
569469       1    1       1      1     1     1      1     1     1     1     1      1    0    0    2
80777        1    1       1      1     1     1      1     1     1     1     1      0    1    1    1
7216         1    1       1      1     1     1      1     1     1     1     1      0    1    0    2
31           1    1       1      1     1     1      1     1     1     1     1      0    0    1    2
227981       1    1       1      1     1     1      1     1     1     1     1      0    0    0    3
24817        1    1       1      1     1     1      1     1     1     1     0      1    1    1    1
2450         1    1       1      1     1     1      1     1     1     1     0      1    1    0    2
125632       1    1       1      1     1     1      1     1     1     1     0      1    0    0    3
10700        1    1       1      1     1     1      1     1     1     1     0      0    1    1    2
1611         1    1       1      1     1     1      1     1     1     1     0      0    1    0    3
73525        1    1       1      1     1     1      1     1     1     1     0      0    0    0    4
16763        1    1       1      1     1     1      1     1     1     0     1      1    1    1    1
1258         1    1       1      1     1     1      1     1     1     0     1      1    1    0    2
2            1    1       1      1     1     1      1     1     1     0     1      1    0    1    2
29525        1    1       1      1     1     1      1     1     1     0     1      1    0    0    3
7586         1    1       1      1     1     1      1     1     1     0     1      0    1    1    2
550          1    1       1      1     1     1      1     1     1     0     1      0    1    0    3
12569        1    1       1      1     1     1      1     1     1     0     1      0    0    0    4
1507         1    1       1      1     1     1      1     1     1     0     0      1    1    1    2
70           1    1       1      1     1     1      1     1     1     0     0      1    1    0    3
3970         1    1       1      1     1     1      1     1     1     0     0      1    0    0    4
741          1    1       1      1     1     1      1     1     1     0     0      0    1    1    3
45           1    1       1      1     1     1      1     1     1     0     0      0    1    0    4
2554         1    1       1      1     1     1      1     1     1     0     0      0    0    0    5
```

On the Figure 2.2 below, the missing values are in red and the observed values in blue.

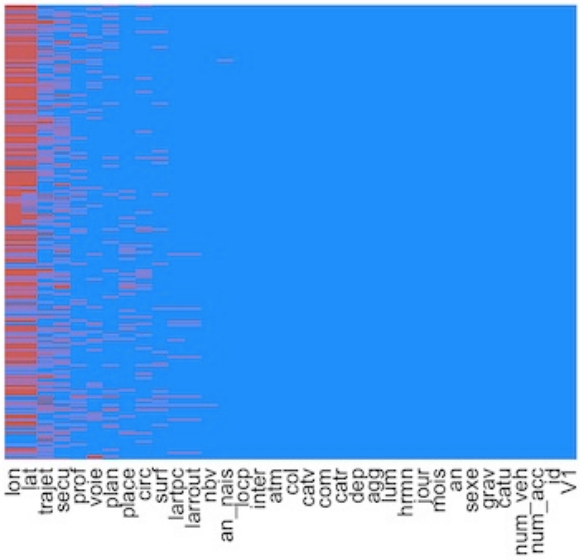


Figure 2.2: Missing values vs observed

This graph allowed us to see that it was impossible to do anything with the longitude and latitude variables. Initially, we wanted to make a cartography but considering the lack of true data of these 2 variables, we had to give up the idea. Thus, we made one in the second chapter, using the variable related to the French departments.

Moreover, the 2 previous images show us that the missing values were not related to each other.

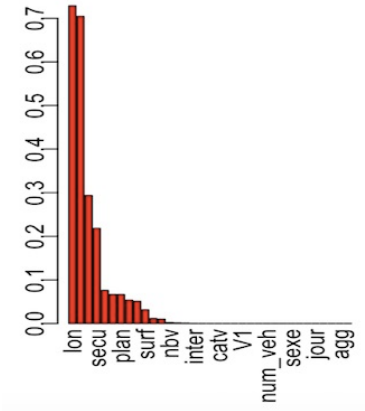


Figure 2.3: Histogram which represents the proportion of missing values

A visual representation thanks to the VIM package could be made to better analyze the missing values. Through this approach, the situation looked a bit clearer, in our opinion.

2.3.3 STEP 3: Our KNN

To be able to do our KNN, we had to create a data frame with our data.

As we said before, it seemed reasonable to delete the characteristics lon and lat before doing our KNN.

We first made a table of 1 milion observations/individuals, to see if the method worked with our 4GB RAM computer (it could not support the 2 millions). Once the method worked, we launched the code on the complete dataset on a computer of 8GB of RAM. In order to implement this method, it was necessary to split the large set of data in 2, then split each of them again 20 times randomly, and apply to each subset the KNN method. Each sub-table still possessed enough individuals, so that we always had a fairly efficient KNN.

Thanks to another function "imput-knn", it was possible to make regressions on each column, but the estimated values were not satisfaying enough, relatively to the invested time .

If we had had a powerful computer at our disposal, it would have surely not been necessary to divide the table to run the KNN.

In order to compare the two methods, we created missing values in our dataset and compared the results of both. We looked at the method that was the speeder, and the closest to the truth. This way, we could select the most efficient between the imput-knn and the KNN of the VIM package: we chose this latter.

Obtaining the results for our complete table of 2 millions individuals took more than 3 days. To improve the speed, we could have run the code on different cores of the computer.

For our KNN, we chose k=3 (which the default value for k). In general, we notice that it is interesting to take the square root of the number of individuals. In our case, the value of the K would have been too large and we did not have the required equipment.

2.3.4 STEP 4: Check the imputations

By making histograms (barplot) and piechart (pie)[2], we checked that the imputation of the missing values did not modify the overall results.
It was also confirmed by the fact that the percentage of the majority of variables did not change significantly.

[2]The codes are available in the Annexes part

Chapter 3

Data visualization related to the accidents in France

3.1 Exploration and visualization of the data

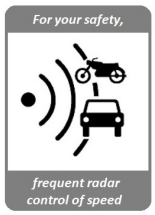
3.1.1 Introduction

For a long time, France has had the reputation of the most dangerous country in Western Europe in terms of road safety, but since 1972, public authorities have looked into this issue and have therefore carried out preventive actions throughout the whole territory.

For example, in 1972, large measures for the control of blood alcohol, speed limitation and belt wearing are adopted. Also, in 1989, "the white paper on road safety" is published. It prescribes various measures to strengthen the training of drivers and the effectiveness of the control-sanction system, some of which have been adopted during the 1990's.

In 2002, the proliferation of controls, the aggravation of sanctions and the creation of a probationary driving license, were organized.

The goal of these actions was to reduce the number of road accidents, by encouraging citizens to respect the road's code, to adopt preventive behaviors (not to drive after drinking, for instance), and in particular, to control their speed.



3.1.2 Evolution of the number of road accidents in France

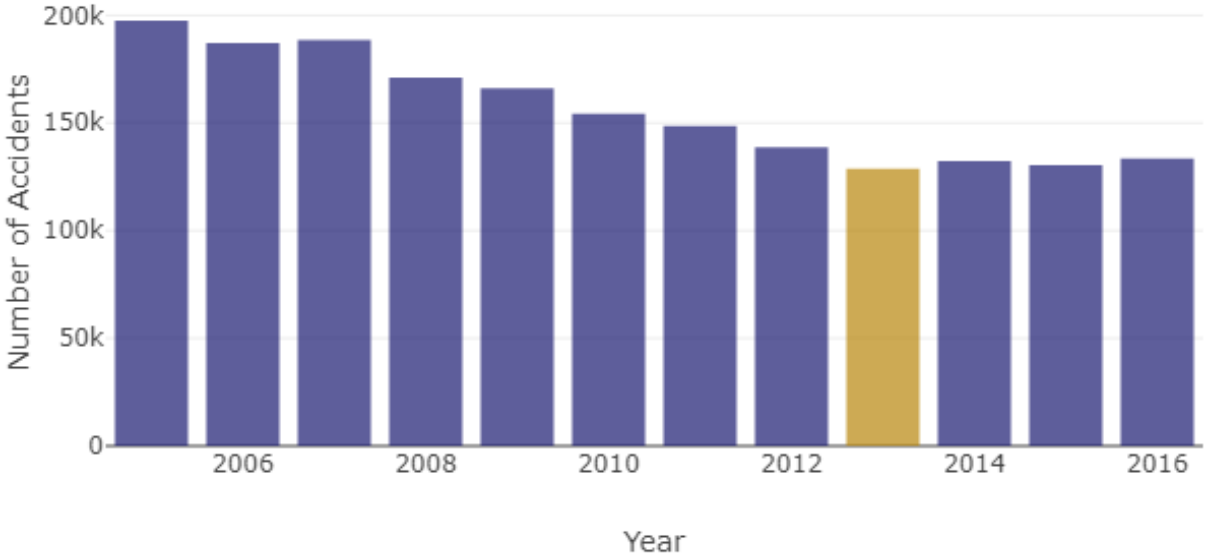


Figure 3.1: Evolution of the number of accidents in France between 2005 and 2016

It can be seen that the number of accidents decreased between 2005 and 2013. This could be explained by the installation in 2004, of the first automatic radars.

However, the lowest number of accidents in the studied period was recorded in 2013, with a decrease of 7.16% compared to the previous year. This could also be due to the installation of RMNG (new generation of mobile radars).

However, it can be seen that this number tends to increase again slightly, from 2014 to 2016. The reasons are not well-known yet.

Along with the gradual installation of radar, the improve of the quality of cars over time may be a factor to consider in this general decline in accidents that we observe here.

3.1.3 Data related to the localisation of accidents

The number of traffic accidents in France has been decreasing since 2005.

However, what about the distribution of these accidents in France and in the French Overseas Territories?

Over these 11 years, there was 839,985 accidents in France.

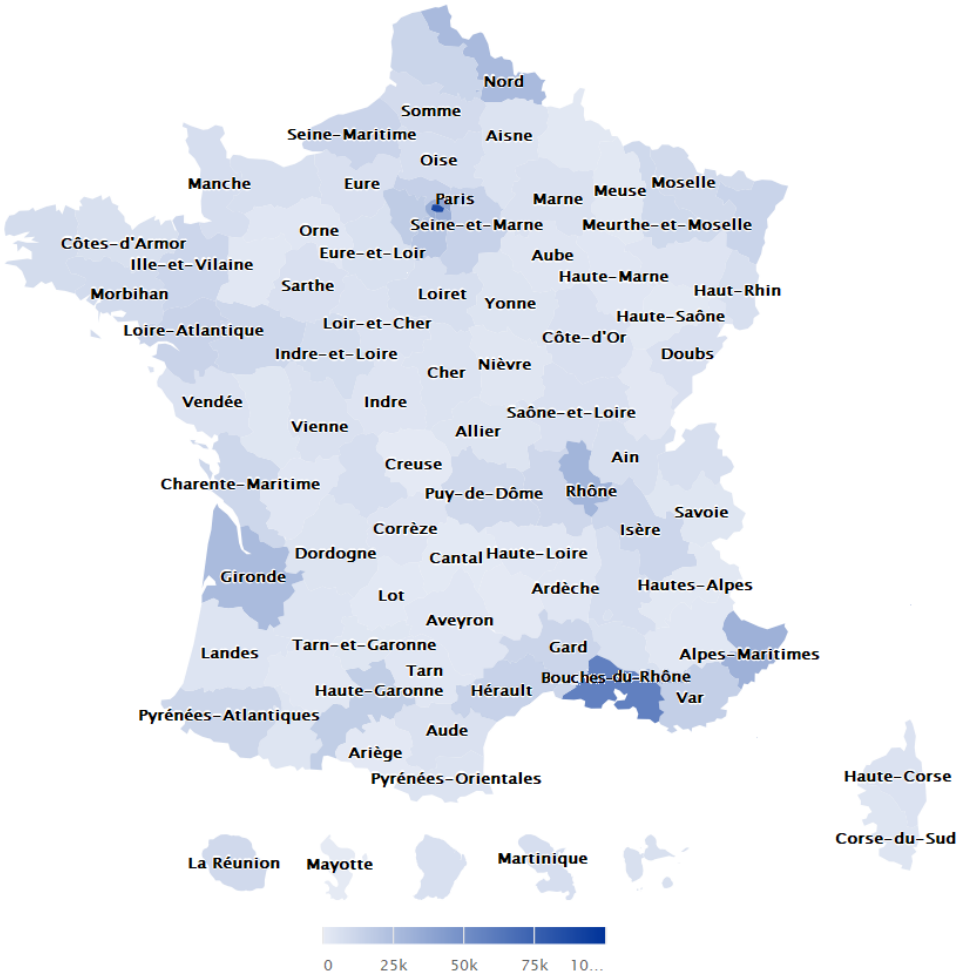


Figure 3.2: Repartition of the accidents in France and in the French Overseas Territories

In our dataset, we had information on the location of the accident, such as longitude and latitude and the department number according to the INSEE code.

As explained above, the longitude and latitude variables were removed from our dataset because they contained more than 50% of missing values, so it was impossible for us to make a map with these data: some accidents would have been removed from the calculation and our interpretation could have been distorted.

Thus, we used the department number to do this study, as it did not contained any missing values .

We can clearly see that the most accidents occurred in Paris, with 87,482 accidents between 2005 and 2016. The second most accident-prone department is the Bouche-du-Rhône one, with 52,188 accidents over these 11 years.

We can notice that the common point between these two departments is that they each have the two largest cities of France, and therefore, the influx of people is greater, which of course increases the number of accidents.

Also, the complexity of the road network in big cities could lead to this observation too.

Next, we find the departments of the Ile de France region, with 33,810 accidents in Seine-Saint-Denis, 31,261 accidents in Hauts-de-Seine, 29,389 accidents in Val-de-Marne, etc.

Other departments, such as the Alpes-Maritime, the Rhone, the North, and the Gironde are among the most accident-prone departments. Here also, the numerous population would be the cause, because these departments are of the most inhabited (or populated) after the capital and Marseille.

On which type of roads are there the most accidents?

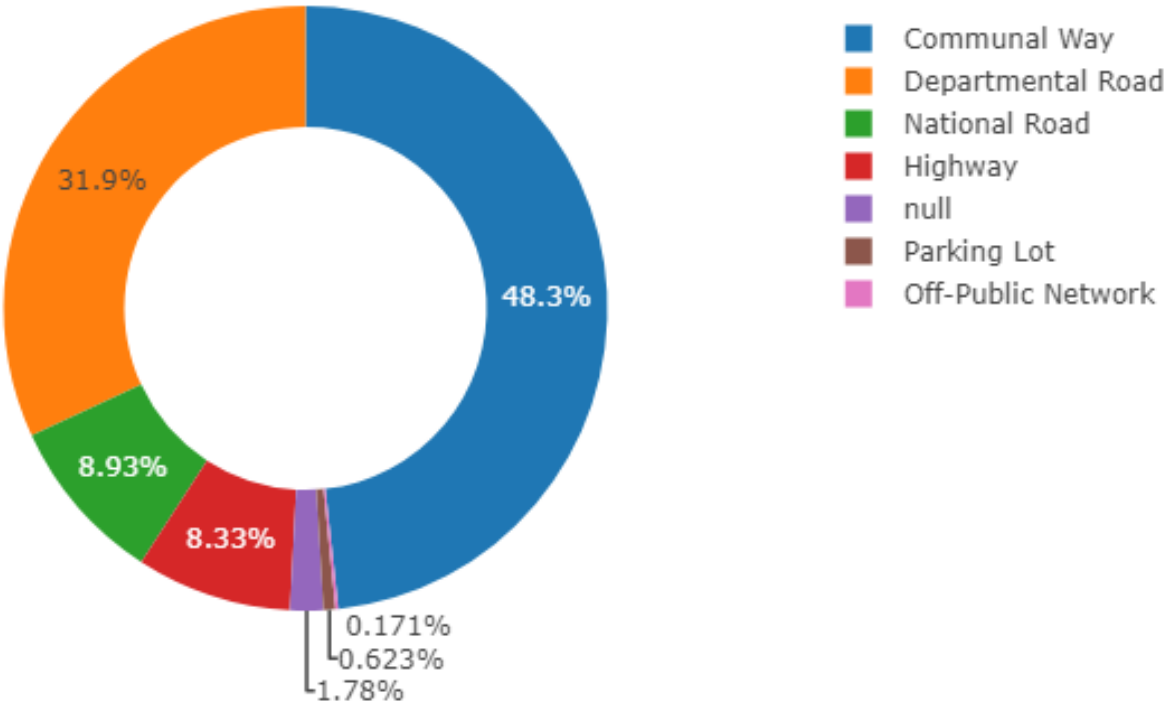


Figure 3.3: Percentage of accidents according to the type of the road

We can see that 48% of car accidents in France occur on municipal roads, that is to say, in urban areas. Almost 32% of accidents occur on departmental roads, and around 9% on national roads.

Thus, the secondary roads (communal and departmental) would expose us to the risk of having an accident more than highways for example, where the quality of the roads is better, the security is reinforced with a better luminosity, and anti-drowsiness strips are installed since 2012.

Do the most serious accident occur in or out an agglomeration?

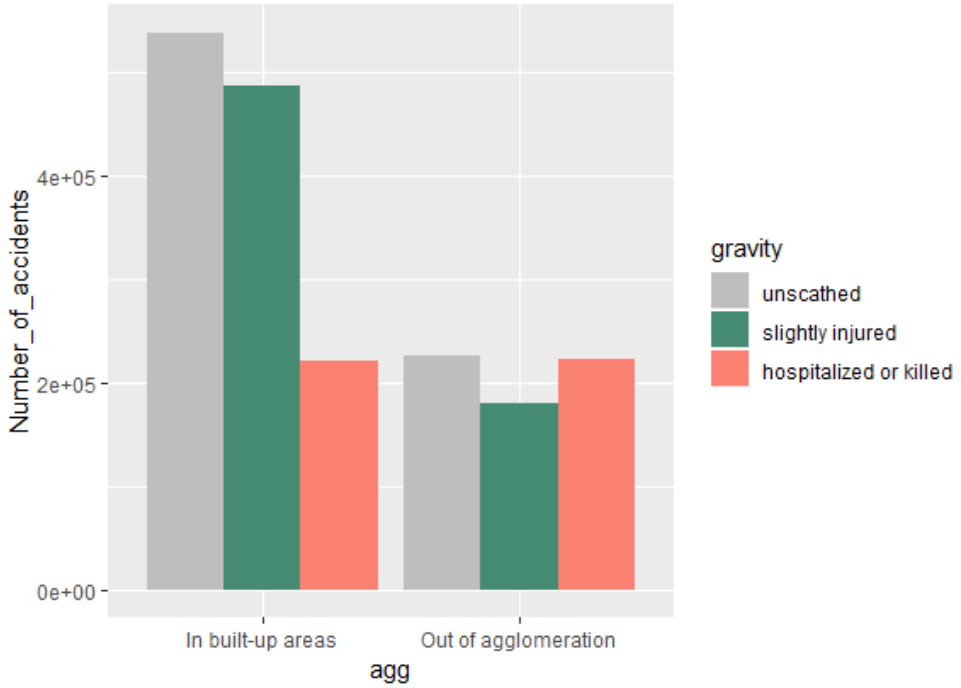


Figure 3.4: The severity of the accidents according to the area

It can be seen that the number of accidents is significantly higher in urban areas than outside.

People who have been unscathed or slightly injured are twice as numerous in urban areas than outside. Nevertheless, it is interesting to note that the number of hospitalized or killed people is similar in both geographical areas.

In urban areas, as the road network is more complex, accidents are more frequent. Moreover, because of the limitation of the speed of 30 or 50km / h, people that are involved in an accident are mostly unscathed or slightly injured.

On the other hand, out of agglomerations, the speed is limited to 90km / h, and so involved people often get more serious injuries. This could probably be due to a more important shock.

Moreover, as mentioned above, most accidents occur on secondary roads (communal and departmental ones). Thus, the fate of people who are involved in an accident could also depend on the time of intervention of the relief, which would be a little longer in such areas, where the access is more difficult.

Another plausible explanation would be that the risk of over-accident (multiple collisions) is higher outside an agglomeration. This type of accident, which therefore involves more people, would then cause more damages.

Which type of intersection should be the most feared?

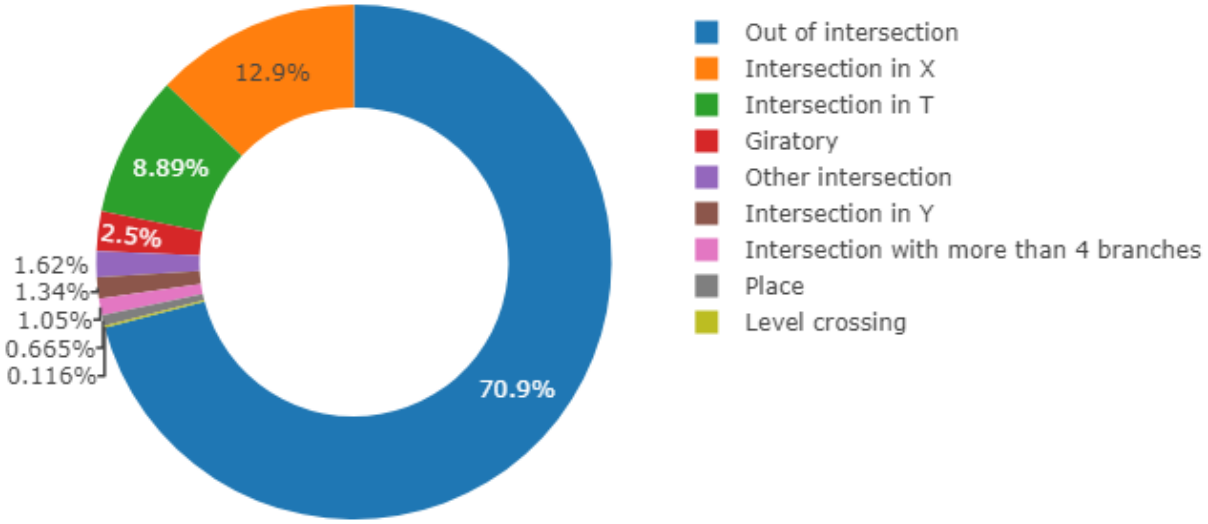


Figure 3.5: Percentage of accidents according to the type of intersection

It has been statistically shown that most accidents that occurred in an intersection are caused by driver's errors, and this could be avoided with some assistance systems of cars.

First, BMW created some, for example the "Left Turn Assist" which consists in showing a red pictogram to alert the driver of the arrival of a vehicle in the opposite direction. The aim is to avoid collisions when the driver wants to turn left (by crossing the opposite lane).

Also, the "Intersection Assistant" is a sound and light signal that alerts the driver, when approaching an intersection, of the arrival of a car on his left or right.

Here, will be considered accidents that occurred out an intersection, and in intersections in a X, T, or Y shape, or in a roundabout, etc.

From the graph, we can see that 70% of the accidents occurred outside an intersection. This could be explained by the fact that when approaching an intersection, drivers are more attentive to what happens on the road, since they detect a potential danger.

On the other hand, for example, on a straight departmental road without any intersection, drivers will more "flutter", look for something in their bag, or touch the radio station etc. Then, in these moments of inattention, the danger is more present.

Among the different types of intersections we have in our data, the intersection in a X shape appears as the most risky one, with almost 13% of the accidents.

8.89% of the accidents occurred in a T-shaped intersection.

3.1.4 Data related to the weather conditions

Which weather causes the least accidents?

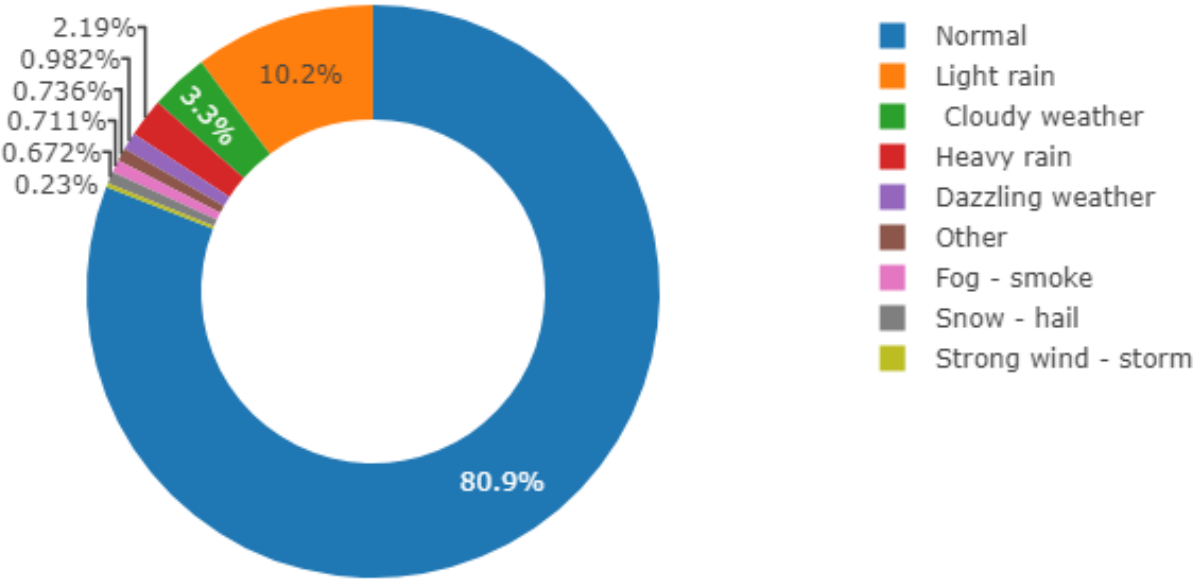


Figure 3.6: Percentage of accidents according to meteorologic conditions

We can see that it is when the weather is nice that 80% of accidents occur. Drivers tend to be less vigilant when the visibility is good.

We could therefore believe that it's in summer that there are the most accidents, since at this time of year, people go on vacation and take their car to make longer journeys.

Thus, tiredness could be a determining factor in the increase of the number of accidents when the weather is good.

We also note that only 10% of the accidents occurred when it was raining.

Is it safer to drive during the day or by night?

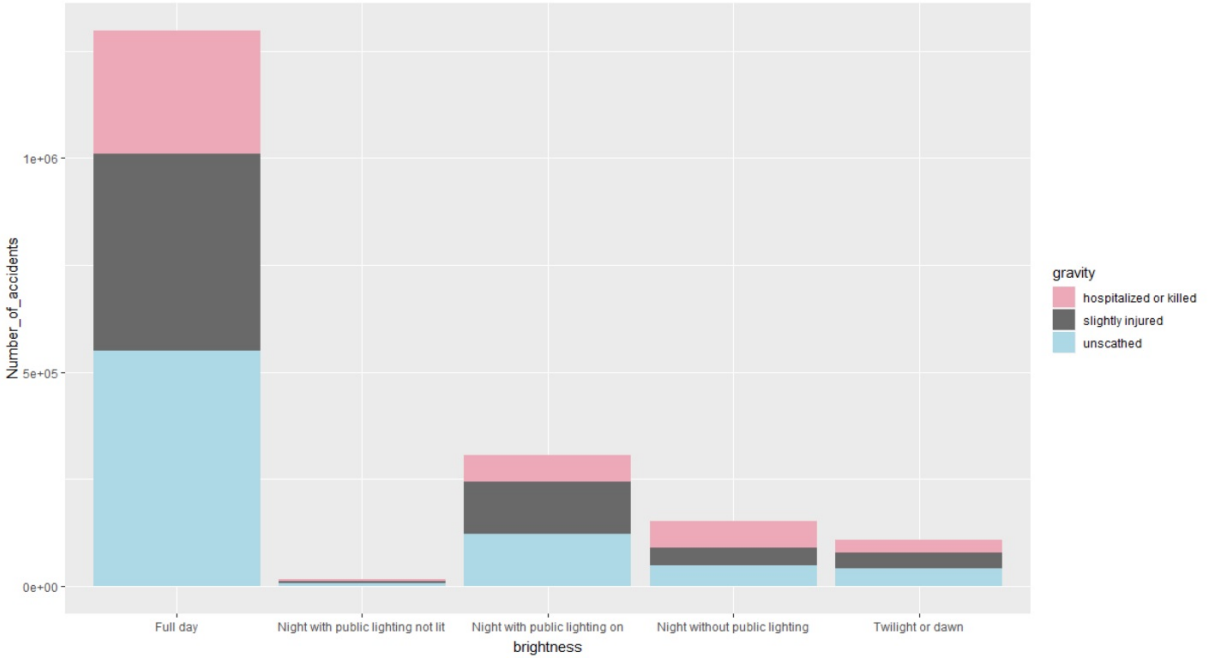


Figure 3.7: Number of accidents according to the luminosity of the road

We can see that accidents mainly occurred during the day.

It is interesting to note that among those who occurred in the evening, there are more accidents when the street lights are switched on than when they're off.

Indeed, when there is no light, people will be more vigilant and this would tend to reduce the risk of accidents.

However, although they are fewer, nighttime accidents on a dark road are not the least in terms of injuries: according to the graph, the number of people who were killed or hospitalized is higher.

3.1.5 Data related to the date

Which month is the safer?

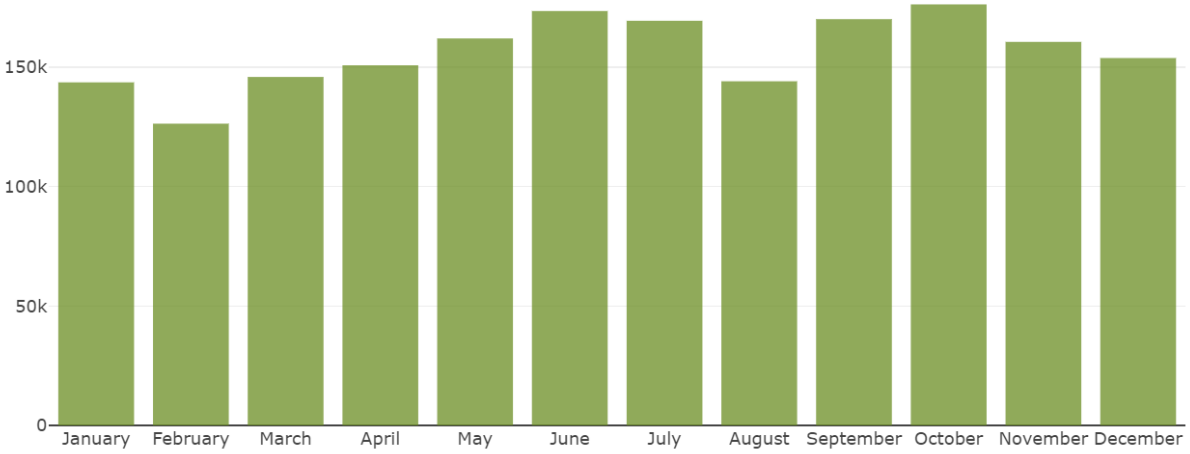


Figure 3.8: The number of accidents per month, in average

Between 2005 and 2016, the month with the highest number of accidents is October, with 176,233 million accidents.

Our study also reveals that it's in February, with 126,313 million accidents, that it would be the safest for people to drive.

Pedestrians are also less involved in an accident during this month.

This could be justified by the fact that at this time of the year, people drive less than in the summer period, and so, this could reduce the risk of accidents.

Moreover, it is well-known that in February, in France, there often is snow on the roads.

We could also imagine that people are more cautious because of these bad conditions, and also, that there are fewer pedestrians in winter than in summer since it is way colder.

At which moment of the day do serious accidents mostly occur?

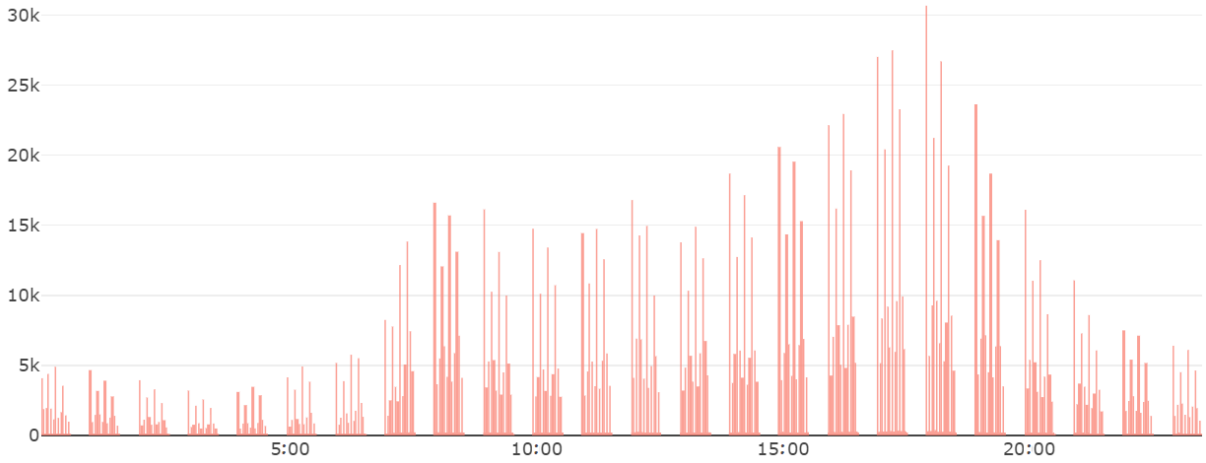


Figure 3.9: The number of accidents during an average day

Regarding the moment of the day where there is the most accidents, we found out it's in rush hours, at around 6pm, that there are the most accidents, with more than 30 million accidents against less than 5 million accidents in the early afternoon.

This is of course due to the fact that in rush hours, the number of cars and people on the road increases, and this increases the risk of accidents.

In addition, at the end of the day, the attention of drivers tends to decrease, because a tired driver is less focused and more disturbed by the urge to return home. This could therefore be a cause of this high occurrence of accidents.

3.1.6 What our data can also teach us

The reasons why people are on the road, according to their age and sex

Here, we want to study the distribution of the reasons for people to drive, according to their age and sex.

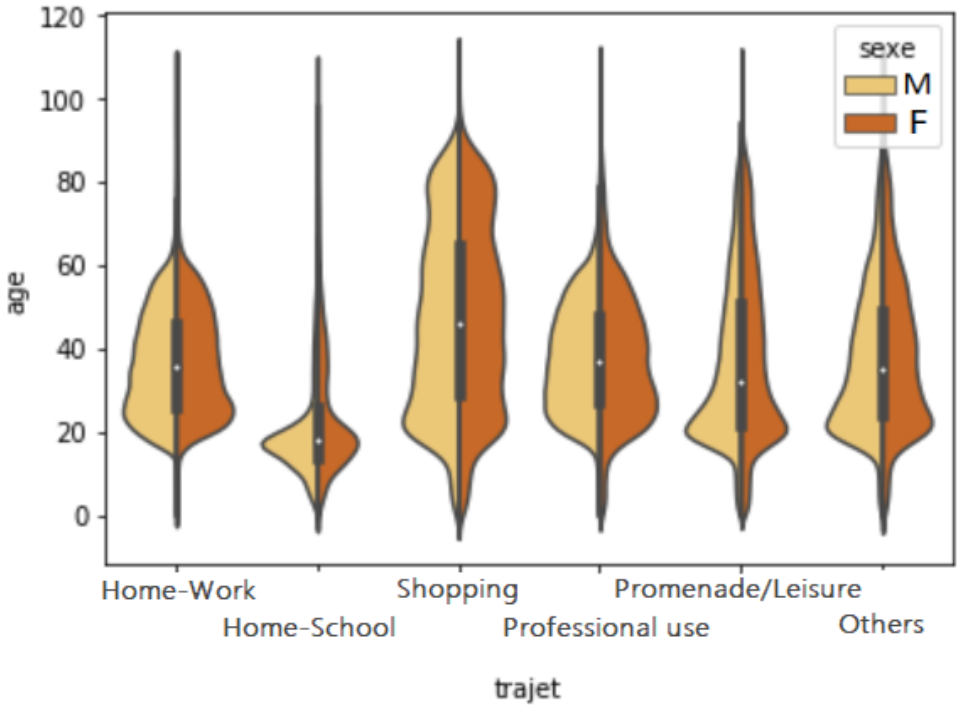


Figure 3.10: Repartition of accidents for each reason of travelling, according to the sex and the age of drivers

This graphic shows that when they had an accident, men and women were mostly traveling for similar reasons.

We can see that most of people who had an accident when they were on the road between home and school are from 10 to 25 years old, and that men are more affected by these accidents than women.

However, there are more women in the tail of the distribution, which implies that women continue to have accidents on this journey up to about 50 years.

We can understand here that it is mainly about women who drive their children to school.

Regarding the accidents during trips for shopping, in contrary to what one might think, 20 years old men are the most concerned.

More generally, the number of accidents was plotted according to the reason of the drivers' journey.

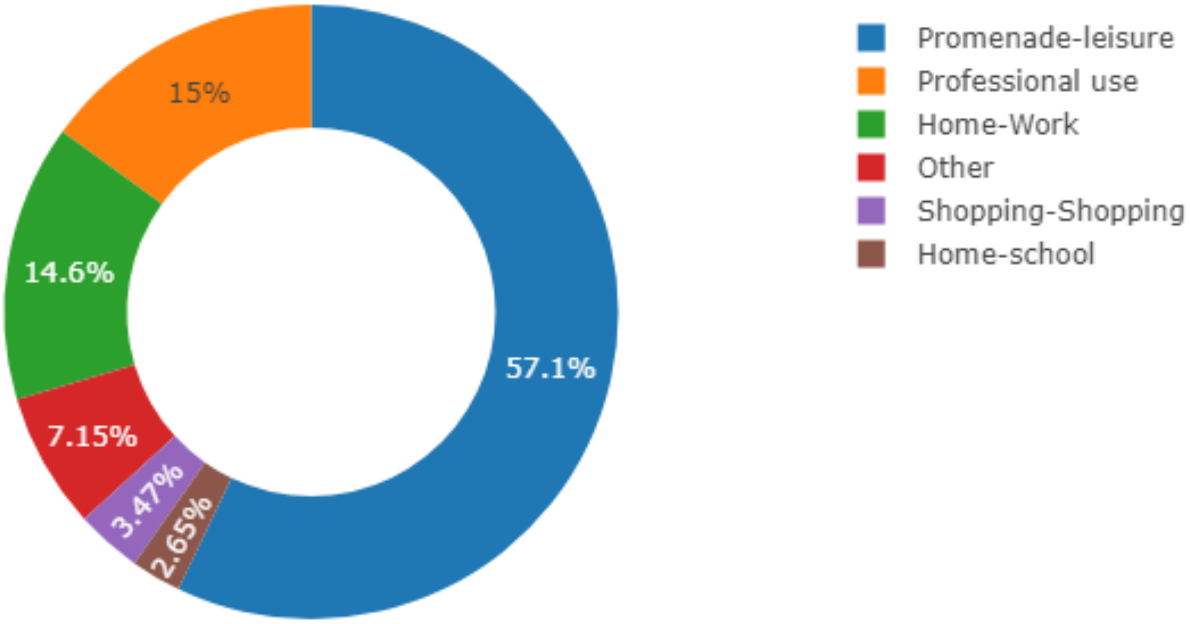


Figure 3.11: Percentage of accidents per reason of travelling

Finally, the most accidents occur when people drive for personal reasons, since there are 57.1% of such accidents in our data. Indeed, they concern the widest age range (15 to 85 years old).

Next, the accidents that occurred when people were driving for professional reasons represent 15% of the accidents. People who are affected by such accidents are mostly from 20 to 50 years old.

When driving for professional reasons, studies showed that people are more dangerous because they are more stressed, and also more in a hurry.

It was noted that 52% of the people are tired when driving for professional reasons, against 45% when driving for other reasons.

This proportion of accidents during work-related trips is closely followed by those who occurred during home-work journeys, with 14.6% of the accidents. The same justifications may be given, as they also relate to work.

Time series visualization

A time series is a set of observations where variables take their values at different times, in an uniform interval (here, the number of accidents per day over several years).

The main objective of the analysis of a time series is to determine trends within these series, as well as the stability of the values and their variation over time. In particular, these trends are often analyzed in the long term to allow forecasts of the evolution of the studied variable to be made.

In these types of data, the time is an important vector, and the order of the events is a key feature of the time series data, as it results in naturally ordered data in time.

For our database, the accidents that occurred form a chronological series with a time vector ranging from the years 2005 to 2016. Thus, here, the studied variable is the number of accidents that occurred each day during this period.

We plotted some graphs showing the evolution of the number of accidents in France from 2005 to 2016, over different periods.

Figure 3.12: Evolution of the number of accidents in France, from 2005 to 2016, over 6 months, one year, 2 years and over the whole studied period.

On the graphs, in addition to the number of accidents represented by the blue curve,we can observe the rolling average (represented by the yellow curve) and variance (represented by the red curve), which represent the average and variance computed on the last p months (where $p = 9$ months in our case) and this will allow us to identify any trend of our variable.

The first graph represents the evolution of the number of accidents over 6 months (in 2016). We see that the largest peak is reached in October.

However, when looking at our time series over one year, on the second graph, we also see an increase in the number of accidents during the summer period, between May and July.

We observe the same phenomenon if we visualize the time series over 2 years, which is represented on the third graph: there are strong increases in June, in July and in October.

When studying the series over the whole studied period, according to the last graph, we can clearly see there is a seasonality, more precisely we can see that, for each year, between October and December, and in May and June, the number of accidents in France reaches a peak, and from January to April as well as in August and September, it reaches a low.

Moreover, by analyzing the rolling average over the whole period, we can also see an overall downward trend in the number of accidents from 2005 to 2016.

Chapter 4

Econometric study of the severity of the accidents

4.1 Theoretical explanations of econometric regressions

4.1.1 Introduction

Econometric regressions are used to study the impact of some parameters onto a variable we want to explain. They can be written as mathematic equations. There exists several types of regressions : linear regressions for linear models, simple probit/logit regressions for qualitative models, ordered logit/probit regressions or also non-ordered logit/probit regressions for mutinomial qualitative models, etc.

Here, we want to study the impact of different variables on the severity of road accidents, so that we can predict the probability for an accident to be more or less serious, depending on these variables.

Econometric regressions are suitable to such studies and that's why we chose this method.

In our model, the severity of the accidents is encoded by the "Grav" variable. Its modalities are 1 if the person is unscathed, 2 if he or she's slightly injured, and 3 if he or she's killed or hospitalized.

When the variable we want to study is a qualitative variable that has more than two modalities, we first determine if there is an order in the modalities. If so, we use an ordered logistic regression, otherwise we use other methods like a multinomial logistic regression, a nested logit etc.

Here, we can clearly see we have more than 2 modalities and that ordering them is meaningful. That's why we chose to do an ordered logistic regression.

4.1.2 The ordered logistic regression

The ordered logistic regression is a generalization of the simple logistic regression. The difference between both is, that in the case of a simple logit, the variable we want to explain only maps 2 modalities (called a dichotomic variable). Besides, an ordered logistic regression is used to explain a polytomic variable, that means it has more than 2 modalities. "Ordered" means that these modalities may be logically ordered.

$$\text{We can write : } Grav_i = \begin{cases} 1 & if\ unscathed \\ 2 & if\ slightly\ injured \\ 3 & if\ hospitalized\ or\ killed \end{cases}$$

Actually, there is a hidden latent variable behind the terms 'unscathed', 'slightly injured' and 'killed or hospitalized'. This unseen variable is defined as :

$$Grav_i* = X_i\beta + u_i$$

But there is no need to explain what this variable corresponds to, as most of the time, it has no real signification. The only thing is that it has to be a continuous variable that affects $Grav_i$.

$$\text{Thus, we have: } Grav_i = \begin{cases} 1 & if\ Grav_i* < \mu_1 \\ 2 & if\ \mu_1 \leq Grav_i* < \mu_2 \\ 3 & if\ \mu_2 \leq Grav_i* \end{cases} \qquad where\ \mu_j\ will\ be\ estimated.$$

Then, we can define the predicted probabilities for $Grav_i$ to be $j = 1$, 2 or 3, as follows, supposing $\sigma = 1$, $\mu_0 = -\infty$ and $\mu_3 = +\infty$:

$$\begin{aligned} P(Grav_i = j) &= P(\mu_{j-1} \leq Grav_i^* \leq \mu_j) \\ &= P(\mu_{j-1} \leq X_i\beta + u_i \leq \mu_j) \\ &= P(u_i \leq \mu_j - X_i\beta) - P(u_i \leq \mu_{j-1} - X_i\beta) \\ &= F(\mu_j - X_i\beta) - F(\mu_{j-1} - X_i\beta) \end{aligned}$$

As we are using a logistic regression, we have : $P(u_i \leq a) = \dfrac{e^a}{1+e^a}$

Thus, we have: $P(Grav_i = j) = \dfrac{e^{\mu_j - X_i\beta}}{1+e^{\mu_j - X_i\beta}} - \dfrac{e^{\mu_{j-1} - X_i\beta}}{1+e^{\mu_{j-1} - X_i\beta}}$

4.1.3 Interpretation of the ordered logistic regression's results

The values of the estimated coefficients β_k cannot be interpreted, as they depend on the normalization of σ. As we are in an ordered logit, all we can say is that if $\beta_k > 0$, our $variable_k$ has a positive effect onto the probability for $Grav_i$ to be equal to its last modality. Obviously, it also means it has a negative effect onto the probability for $Grav_i$ to be equal to its first modality. Also, if $\beta_1 \geq \beta_2$, the variable associated to β_1 has a more important impact onto the severity than the one associated to β_2, provided that both variables are of the same form (two dichotomic variables, for example).

But we cannot tell anything about the probability for $Grav_i$ to be equal to its others modalities. We cannot either interpret the values of β_k, that's why it is relevant to calculate predicted probabilities and the marginal effects.

Here, our 3 modalities are ordered from the least to the most serious injuries. Thus, a positive coefficient means the variable increases the probability for an accident to be more serious. However, if we want to quantify this effects, the only solution is to calculate the marginal effects as shown thereafter.

The marginal effect of the variable X_i, onto the probability $P(Y_i = j)$ is :

$$\frac{\delta P(Y_i = j)}{\delta X_i} = \beta_i(\ f(\mu_j + 1 - X_i\beta) - f(\mu_j - X_i\beta)\ )$$

After that, we are able to interpret the coefficients. For example, if $\beta_k = x$ with $x > 0$, we can say, if the $variable_k$ is a dichotomic variable, that it increases the probability $P(Y_i = j)$ by $x * 100$ points of percentage. Otherwise, if $variable_k$ is a continuous variable, then we will say that when it increases by 1 unit, then the probability $P(Y_i = j)$ increases by $x * 100$ points of percentage. Obviously, a negative coefficient has the opposite effect.

4.2 Data treatment

In order to be more precise in the results of our economic regression, we created another dataset from the first one, without the observations containing missing values.
Thus, for this part, we only studied the 829,575 persons for which we had all the information. We also deleted the pedestrians, as they were still only 16, they wouldn't have been representative.
Thereby we decided to study the severity of the accidents onto the people inside the vehicles, such as drivers or passengers (of cars, moped vehicles, etc.).

Then, we manipulated the modalities of all the variables in order to get at least 5% of the observations in each, except for "secu": we only have 3.62% of "absence of a security system" but we chose to keep it anyway, as it was significative, because if we didn't, we couldn't use the variable and it seemed to be important.
Then, we created as many dichotomic variables as there were modalities in each variable.

```
. tab grav
```

grav	Freq.	Percent	Cum.
1	373,040	44.97	44.97
2	268,722	32.39	77.36
3	187,813	22.64	100.00
Total	829,575	100.00	

```
. tab secu
```

secu	Freq.	Percent	Cum.
0	30,024	3.62	3.62
1	799,551	96.38	100.00
Total	829,575	100.00	

```
. tab driver
```

driver	Freq.	Percent	Cum.
0	60,001	7.23	7.23
1	769,574	92.77	100.00
Total	829,575	100.00	

```
. tab turn
```

turn	Freq.	Percent	Cum.
0	679,319	81.89	81.89
1	150,256	18.11	100.00
Total	829,575	100.00	

```
. tab summer
```

summer	Freq.	Percent	Cum.
0	613,370	73.94	73.94
1	216,205	26.06	100.00
Total	829,575	100.00	

```
. tab coli
```

coli	Freq.	Percent	Cum.
0	56,702	6.84	6.84
1	772,873	93.16	100.00
Total	829,575	100.00	

Figure 4.1: Some examples of our variables

4.3 Application and results interpretation

Intuitively, we can make some assumptions about the results we will get. Considering our variables, we could imagine that a man would be less injured than a woman, if all other variables are held at their means.

We could think that being a driver decreases the probability for the injuries to be more serious, as we often talk about the "death seat", which is the passenger's one, next to the driver.

We can suggest the serevity of injuries would be higher if there was a collision.

Also, we can think a bad weather or bad conditions such as a slippery or darkened road, would lead to more serious injuries.

We're also pretty sure that security systems are essential.

So now, let's see what our data tells us !

```
. mfx, predict(p outcome(3))

Marginal effects after ologit
      y  = Pr(grav==3) (predict, p outcome(3))
         =  .17859555
```

variable	dy/dx	Std. Err.	z	P>|z|	[95% C.I.]		X
driver*	-.0686995	.0015	-45.83	0.000	-.071638	-.065761	.927673
man*	-.0700017	.00084	-83.22	0.000	-.07165	-.068353	.715299
secu*	-.2489892	.0029	-85.91	0.000	-.25467	-.243309	.963808
age	-.0005011	.00002	-24.41	0.000	-.000541	-.000461	38.0672
peak_h~r*	-.0164907	.00067	-24.51	0.000	-.017809	-.015172	.392097
agg*	-.1403894	.00108	-129.69	0.000	-.142511	-.138268	.61119
normal~r*	.0057405	.00124	4.63	0.000	.00331	.008171	.839425
nbv	-.0042107	.00025	-16.73	0.000	-.004704	-.003717	2.18367
flat_r~d*	-.0034188	.00084	-4.06	0.000	-.005071	-.001767	.801372
turn*	.0493128	.00103	47.73	0.000	.047288	.051338	.181124
lartpc	.0000397	.00002	2.64	0.008	.00001	.000069	6.09056
larrout	2.24e-06	.00001	0.41	0.683	-8.5e-06	.000013	65.9468
slippery*	.0188537	.00124	15.21	0.000	.016424	.021284	.198001
coli*	-.0936603	.00185	-50.58	0.000	-.09729	-.090031	.931649
loisir~n*	.0033558	.00115	2.91	0.004	.001094	.005617	.554361
pro_re~n*	-.040335	.00116	-34.91	0.000	-.0426	-.03807	.357234
spring*	.0048357	.00098	4.93	0.000	.002913	.006759	.24496
summer*	.0065789	.00099	6.66	0.000	.004642	.008516	.260621
autumn*	-.0028541	.00092	-3.10	0.002	-.004661	-.001047	.272479
day_li~t*	-.0636317	.00134	-47.44	0.000	-.06626	-.061003	.69508
morn_l~t*	-.0272791	.00156	-17.54	0.000	-.030328	-.02423	.058754
lighte~t*	-.0306911	.00128	-23.89	0.000	-.033209	-.028173	.151645
inters~n*	.0222574	.00128	17.36	0.000	.019745	.02477	.720058
x_inter*	.0290581	.00171	16.99	0.000	.025707	.032409	.125158
t_inter*	.0089508	.00172	5.20	0.000	.005578	.012323	.088471
face_2~l*	.0613348	.00142	43.15	0.000	.058549	.064121	.12719
file_2~l*	-.0164529	.00108	-15.29	0.000	-.018563	-.014343	.138995
cote_2~l*	-.0126106	.00094	-13.39	0.000	-.014457	-.010764	.327465
file_p~l*	-.0345737	.00132	-26.13	0.000	-.037167	-.031981	.064249
plus_m~l*	-.0397629	.00125	-31.70	0.000	-.042221	-.037304	.068169
highway*	-.0219529	.00145	-15.19	0.000	-.024786	-.01912	.106606
nat_road*	.0358153	.00144	24.91	0.000	.032997	.038633	.094102
dep_road*	.0624868	.00092	67.65	0.000	.060676	.064297	.361568
one_wa~d*	-.0143201	.00115	-12.41	0.000	-.016582	-.012058	.169745
two_wa~d*	.0238565	.00106	22.52	0.000	.02178	.025933	.66478
veh_ty~2*	.128966	.00209	61.81	0.000	.124877	.133055	.055391
veh_ty~7*	-.2306665	.00112	-205.13	0.000	-.23287	-.228463	.635656
veh_t~10*	-.1504099	.00062	-243.70	0.000	-.15162	-.1492	.052279
veh_t~33*	.1767889	.00204	86.77	0.000	.172796	.180782	.075411

(*) dy/dx is for discrete change of dummy variable from 0 to 1

Figure 4.2: STATA Regression - Marginal effects

On figure 4.2 is our STATA regression[1], with the marginal effects onto the last modality (being seriously injured). We decided to directy study the marginal effects so that we can interpret the coefficients[2].

[1] All the codes of our computations are available in the Annexes part

[2] The coefficients are on the first red column

4.3.1 Relevance assessment of our model

First of all, we will check that our model is relevant by calculating the number of correct predictions of each modality:

Here are the confusion matrix and the rates of correct predictions we calculated "by hand":

Predicted \ Reality	$Grav_i = 1$	$Grav_i = 2$	$Grav_i = 3$
$Grav_i = 1$	292,449	130,072	43,855
$Grav_i = 2$	63,778	91,897	46,753
$Grav_i = 3$	16,803	46,753	72,332
Total	373,030	268,722	162940

Table 4.1: Confusion matrix

	Rate of correct predictions
unscathed	78.4 %
slightly injured	34.2%
seriously injured	44.4%

Table 4.2: Rates of correct predictions of our model

We can see that the second modality, which corresponds to light injuries, is not predicted well by our model but this does not really matter since we will focus on the first and last ones.

The first modality, being unscathed, is very well predicted. According to the last one "serious injuries", our predictions are not too bad, but they could have probably been better if we had had information about the BAC (Blood Alcohol Content) of drivers, or the use of their mobile phone, etc.

Fortunately, we can finally conclude our model is relatively pretty good. Anyway, as we had no more variables to add, we could not have improved it and so we are glad it is relevant. But if it was not, that would have been a result to consider anyway.

As we consider our model is relevant enough, we will analyse its results.

4.3.2 Analysis of the results

The first thing to do is to look at the column $p >| z |$, as if the value is greater than 0.05, that means the variable is not statistically significant[3]. In other words, that means the variable does not statistically have an impact onto our outcome variable "Grav".

Thus, we can see that all our variables are statistically significant but one : larrout, which is the width of the road. That means that given that all other variables are held constant, the width of the road does not have a significative impact onto the severity of an accident.

We computed the marginal effects onto the last modality of our outcome variable, "hospitalized or killed". Thus, we can interpret like : a positive coefficient x means the variable increases the probability to be hospitalized or killed by x points of percentage.

We computed the predicted probabilities and obtained:

	predicted probability
$Grav_i = 1$	42.65%
$Grav_i = 2$	39.49%
$Grav_i = 3$	17.86%

Table 4.3: Predicted probabilities of the severity of the injuries due to an accident

Thus, at this step, we can say that, given that all of the other variables in the model are held constant:

Being a driver decreases the probability of being hospitalized or killed by 6.87%[4] compare to passengers.

The probability for a man to be hospitalized or killed is lower of 7% than for women.

The chances to be seriously injured are higher when people travel for personal reasons (vacations, shopping etc.) than when they do for any other reason. The reasons that decreases the most the probability to be seriously injured are the professional ones.

A good weather increases the probability for injuries to be more serious by 0.5%. We didn't expect this result, but actually, we could think that when the weather is nice, people are less concentrated, more confident so that they drive faster than if it was raining or snowing.

[3]The second red column

[4]We will write "%", but we actually talk about points of percentage.

Surprisingly, the variable "coli", which corresponds to the fact that there was a collision between at least two vehicles, has a negative coefficient: that means that, given that all of the other variables in the model are held constant, a collision between vehicles decreases the probability for people to be hospitalized or killed.

Moreover, the chances to be seriously injured increase by 9.37% when no collision occurs. We will go further to analyse this surprising result:

We calculated the predicted probabilities for people to be hospitalized or killed with and without a collision:

	everybody	Drivers	Passengers
Collision	17.31%	16.88%	23.6%
No collision	26.68%	26.09%	34.93%

Table 4.4: Predicted probabilities of being seriously injured with and without a collision, in average, for drivers and for passengers

The probability for driver to be seriously injured, given that all the other variables are held at their mean, is higher when there is no collision. The predicted probability to be seriously injured in a collision is 17.31%, whereas it is 26.68% without a collision.

The same phenomenon is observed for passengers, but the predicted probabilities are a little higher, and that's because as we saw, being a passenger increases the probability to be seriously injured compared to drivers (given that all of the other variables in the model are held constant).

The accidents without a collision may be mostly due to driver's behavior: tiredness, BAC (Blood Alcohol Content), or overconfidence, etc. These accidents increase the probability to be seriously injured and we will try to find out why.

However, when there is a collision, different types of collisions don't lead, on average, to the same severity. Let's study which types of collision are the worst by calculating the predicted probabilities.

Predicted probabilities of $Grav_i$ = 1, 2 or 3 for each type of vehicles:

$Grav_i$	2 vehicles : frontal	2 vehicles : from the rear	2 vehicles : by the side
unscathed	33.78%	45.63%	44.93%
slightly injured	42.15%	38.21%	38.53%
seriously injured	24.07%	16.15%	16.54%

$Grav_i$	more vehicles : in chain	vehicles : multiple collisions	other collisions
unscathed	49.07%	50.11%	42.79%
slightly injured	36.56%	36.03%	39.43%
seriously injured	14.37%	13.87%	17.78%

Table 4.5: Predicted probabilities of the severity of the injuries by type of collisions

We can see that a frontal collision between 2 vehicles leads to a probability of 33.78% of being unscathed, 42.15% of being slightly injured and 24.07% of being hospitalized or killed. Compared to the other types, we can conclude the most dangerous collision is when two vehicles collide frontally.

We also see that among all the types of collision, the ones that increase the most the probability to be hospitalized or killed are those between two vehicles or the other types of collision. Thus, we can suggest the more vehicles are involved, the more the probability of being seriously injured decreases.

This is coherent with the fact that accidents without collision (so, with only one involved vehicle) tend to be more serious than the others.

We could imagine that's because when there is a collision between vehicles, the shock is partially "absorbed" by the car bodies and so, people are less injured. But this would have to be scientifically proved.

Moreover, collisions are supposed to trigger the airbags so this could help decrease the severity of collisions. Besides, unfortunately, most accidents without collision don't, so it could explain the difference between both types of accidents.

- Next, we will analyse the role of security systems:
From figure 4.2, we can say that using a security system decreases the probability to be hospitalized or killed by around 24.9%, which is not negligible.

But, how essential are they for our safety ?

We calculated the predicted probabilities of being unscathed, slightly injured and seriously injured with and without the use of a security system:

$Grav_i$	With a security system	Without any security system
unscathed	43.77%	18.19%
slightly injured	39.03%	39.71%
seriously injured	17.2%	42.1%

Table 4.6: Predicted probabilities of the severity of the injuries with and without using a security system

Given that all of the other variables in the model are held constant, the predicted probability of being unscathed while using a security system is 43.77%, against 18.19%.

The use of a security system does not decrease a lot the probability of being slightly injured (it decreases it by less than 1%). Without any security system, the probability to be seriously injured is around twice higher than with one.
Then, we can affirm that security systems do have a vital role in the people's safety.

Finally, we will study the severity of the injuries for people using different types of vehicles: From figure 4.2, we can say types 2 and 33 are the ones that lead to the most serious injuries, compared to all the other types. Types 7 and 10 are the least dangerous ones.

Let's remind that a vehicle of type 2 corresponds to Moped vehicles ($< 50cm^3$) , type 7 corresponds to cars, and type 10, to vans (like commercial vehicles). We don't know the meaning of type 33, so we will just call it "type 33" for now and try to guess what type it corresponds to.

$Grav_i$	Moped vehicles	cars	vans	type 33	other types
unscathed	13.13%	55.91%	63.13%	10.72%	23.82%
slightly injured	35.18%	32.78%	28.24%	31.89%	42.09%
seriously injured	51.69%	11.31%	8.63%	57.39%	34.09%

Table 4.7: Predicted probabilities for each type of vehicles

Type 33 and moped vehicles are the most dangerous types of vehicle, with a probability of respectively 57.39% and 51.69% of being killed or hospitalized, given that all of the other variables in the model are at their mean. Besides, vans are the safest vehicles type with 63.13% of chances to be unscathed.

We could then suppose that type 33 correponds to motorbikes, as they seem to be similar to moped vehicles but more dangerous: moped vehicles have a speed limit, whereas motorbikes don't.

In order to check this, we thought that by computing how many passengers there are on these vehicles, we could get an indication of the type of vehicle. If there are not more that one or two passengers (assuming some motorbikes do have 3 seats), then we can conclude these vehicles of type 33 may actually be motorbikes.

We thus sorted our dataset by the accident id so that all passengers of a same vehicle are ordered. Then, we only keeped the vehicles of type 33 and we calculated how many passengers were on each vehicle.

We found out there was not more than one passenger on each of these vehicles[5].

That seems to be coherent with our suggestion and so we can confirm that vehicles of type 33 may be identified to motorbikes.

[5]The code that computes this is available in the Annexes part.

Chapter 5

Conclusion

Even though the number of accidents has been decreasing for these 11 years, there still are a lot of accidents.

We saw that most of accidents occur when the danger does not seeem to be present. For example, there are more road accidents when the weather is nice than when it's raining, most of accidents occur out of an intersection, or when the road is corretly lightened.

Most of our studies result in the same conclusion: actually, the most accidents do occur when we think the road is safe, whereas when it seems to be dangerous, much less accidents do.

Thus, to decrease the number of accidents, our advice would be not to feel so confident when the road seems to be safe, because it often leads to a too high speed, a lack of attention, or to an overconfidence of the driver, which will increase the probability of having an accident.

But despite our advice should decrease the number of accidents, there will still always be some accidents and so, for the accidents that still occur, we focused on their severity.

We found out that many factors do have an impact onto the severity of accidents and we are now able to give some advice to decrease the probability to have serious injuries when having an accident:

The first thing is that everybody should always use the security systems, such as a helmet for bikers, or a security belt for cars users.

Next, people should pay more attention when riding a moped vehicle or a motobike, as injuries are more serious than with other types of vehicles.

Drivers should become aware that if they have passengers, the danger for them is higher.

People shouldn't be so confident when the weather is nice, because on average, the worst accidents don't occur when it's raining.

Being alone on the road should not justify a higher speed : the speed limits should always be respected, as a simple collision does not result in the worst injuries. Actually, those ones are, on average, mostly as a result of a "without collision" accident.

Obviously, our data does not allow us to talk about speed, the use of a mobile phone or alcohol directly, but we all know they are one of the main factors that lead to accidents. Thus, some other advice that everybody is (hopefully) aware of, would be not to drive after having drunk, to limit the speed even when the road seems to be safe, and not to use a mobile phone while driving.

Chapter 6

Annexes

6.1 Project management

When we found a subject we all were fond of, we asked our referent teacher, Mr. CLOT, if it was a good subject and if he thought its realization was feasible. Once he told us it was, we started trying to manipulate our data but we encountered a lot of difficulties. That's why we asked for his help many times from the beginning.
He answered our questions and then we began to manipulate the data.

There was no leader, we all talked together to find the best solutions. We first defined the structure of our project and at the beginning, each of us was free to work on every part of the project, so that we could all become acquainted with the whole project.

As we always were all connected to a private online group conversation, we could know when someone was working on something and so, we adapted our own works. We told the others each time we finished working and we shared our work on a drive so that they could read, understand and continue it when they wanted.
Thus, each of us naturally found the part she preferred so we defined together who would be working on what part.

When we started our own part of the project, we did it step by step: we began the work and we asked our teachers if they thought we were doing the right thing before moving on. If so, we went on, or we changed what went wrong and after that we continued, considering the remarks they made.
By doing this, we lost less time than if we had done the whole work and then realized we had done something wrong.

Also, we met once or twice a week for a recap, to explain to the others what we had done, to help each other if we had encountered difficulties and to agree on what had to be done next.

We were taught these methods with Mr. ALMENDRA.

6.2 Codes

Code: Merging our datasets (SAS)

```
/* Importation of "caracteristics" table */
PROC IMPORT OUT= WORK.caract
        DATAFILE= "C:\Users\saraa\OneDrive\Bureau\Cours_ISFA\TER\caracteristics.csv"
        DBMS=CSV REPLACE;
     GETNAMES=YES;
     GUESSINGROWS=4000;
     DATAROW=2;
RUN;
/* Importation of "Places" table */
PROC IMPORT OUT= WORK.Places
        DATAFILE= "C:\Users\saraa\OneDrive\Bureau\Cours_ISFA\TER\places.csv"
        DBMS=CSV REPLACE;
     GETNAMES=YES;
     GUESSINGROWS=4000;
     DATAROW=2;
RUN;
/* Importation of "Users" table */
PROC IMPORT OUT= WORK.Users
        DATAFILE= "C:\Users\saraa\OneDrive\Bureau\Cours_ISFA\TER\users.csv"
        DBMS=CSV REPLACE;
     GETNAMES=YES;
     GUESSINGROWS=4000;
     DATAROW=2;
RUN;
/* Importation of "vehicles" table */
PROC IMPORT OUT= WORK.vehicles
        DATAFILE= "C:\Users\saraa\OneDrive\Bureau\Cours_ISFA\TER\vehicles.csv"
        DBMS=CSV REPLACE;
     GETNAMES=YES;
     GUESSINGROWS=4000;
     DATAROW=2;
RUN;
/* Sorting the data by the accident number*/
PROC SORT DATA=Caract;
   BY Num_Acc;
RUN;
PROC SORT DATA=Places;
   BY Num_Acc;
RUN;
/* Sorting the data by the accident number and then by the vehicle number*/
PROC SORT DATA=Users;
   BY Num_Acc num_veh;
RUN;
PROC SORT DATA=Vehicles;
   BY Num_Acc num_veh ;
RUN;
/* We merge Users and Caracteristics tables */
DATA fusion_U_C;
        MERGE Users Caract;
        BY Num_Acc;
RUN;
/* We merge Users and Places tables */
DATA fusion_U_P;
        MERGE Users Places;
        BY Num_Acc;
RUN;
/* We merge Users and Vehicles tables */
DATA fusion_U_V;
        MERGE Users(IN=A) Vehicles;
```

```
        IF A;
        BY Num_Acc num_veh;
RUN;
/* We merge all of our 3 new tables (the ones we obtained by merging users with the others)*/
DATA fusion;
        RETAIN ID Num_Acc num_veh place catu grav sexe trajet secu locp actp etatp an_nais an mois jour
hrmn lum agg int atm col com gps lat long dep catr voie v1 v2 circ nbv pr pr1 vosp prof plan lartpc larrout
surf infra situ env1 senc catv occutc obs obsm choc manv;
        MERGE fusion_U_C fusion_U_P fusion_U_V;
        BY Num_Acc;
        ID = _N_;
        RENAME int = inter;
        RENAME long = lon;
RUN;
/* Our table "fusion" contains all the variables from our 4 initial tables. We sort it by Num_Acc, then by
Num_vehicle and then by Place so that
we can clearly see : which accident, which person, and where this person was */
PROC SORT DATA=fusion;
        BY Num_Acc num_veh place;
RUN;
/* Now we can export our final table as tables_fusionnees.csv */
PROC EXPORT DATA= WORK.FUSION
    OUTFILE= "C:\Users\saraa\OneDrive\Bureau\Cours_ISFA\TER\tables_fusionnees.csv"
    DBMS=CSV LABEL REPLACE;
        PUTNAMES=YES;
RUN;
```

Code: Data mining (STATA, R)

```
###We delete the variables we will never use, on STATA
import delimited "C:\Users\saraa\OneDrive\Bureau\Cours_ISFA\TER\tables_fusionnees.csv"
drop actp etatp gps v1 v2 pr pr1 vosp infra situ env1 senc occutc obs obsm choc manv num adr na
export delimited using "C:\Users\saraa\OneDrive\Bureau\Cours_ISFA\TER\test_stata.csv", replace

#We load the exported table from STATA on R
newtable <- read.csv("test_stata.csv", sep = ",", header = TRUE)
as.data.frame(newtable)
library(ade4)

#We replace for each variable the characters for missing variables by "NA" (some already are NA, but some
are 0)
newtable$inter<-replace(newtable$inter, which(newtable$inter==0),NA)
newtable$circ<-replace(newtable$circ, which(newtable$circ==0),NA)
newtable$prof<-replace(newtable$prof, which(newtable$prof==0),NA)
newtable$plan<-replace(newtable$plan, which(newtable$plan==0),NA)
newtable$surf<-replace(newtable$surf, which(newtable$surf==0),NA)
newtable$trajet<-replace(newtable$trajet, which(newtable$trajet==0),NA)
newtable$secu<-replace(newtable$secu, which(newtable$secu == 0),NA)
newtable$lat<-replace(newtable$lat, which(newtable$lat== 0),NA)
newtable$lon<-replace(newtable$lon, which(newtable$lon == 0),NA)
#for secu, we need 2 numbers and the ones ending with "3" are the "unspecified" ones (missing)
#the ones beggining with 5, 6, 7 or 8 are not defined
#the maximum we can have is 92 (other - absent)
newtable$secu <- as.integer(newtable$secu)
newtable$secu <- replace(newtable$secu, which(newtable$secu%%10 == 3),NA)
newtable$secu <- replace(newtable$secu, which(newtable$secu<11),NA)
newtable$secu <- as.integer(newtable$secu)
newtable$secu <- replace(newtable$secu, which(newtable$secu%%10 == 0),NA)

#We create a variable "age" (age of the person)
age <- (newtable$an + 2000)-newtable$an_nais
newtable <-cbind(newtable,age)
#We delete the variables lat ans lon
newtable <- newtable[,c(1:21,24:35)]
```

```
#We export this new table
write.csv(newtable, "table_NA.csv")
```

Code: Vizualisation of the messy data

```
nrow(completavecvm) #observe the number of individuals

#Histogram of each column to compare before / after imputation of missing values
barplot(table(completavecvm$Colonne)
barplot(table(completsansvm$Colonne)

pie(table(completavecvm$Colonne)
pie(table(completsansvm$Colonne)

#Replace the missing values by the average of the column (change Column by the name of each column)
completavecvm$Colonne[is.na(completavecvm$Colonne)] <- mean(completavecvm$Colonne,na.rm=T)

#Replace the missing values by the median of the column (change Column by the name of each column)
completavecvm$Colonne[is.na(completavecvm$Colonne)] <- median(compeltavecvm$Colonne,na.rm=T)

#Knn on 2 million observations
library(data.table)
library(VIM)
getwd()
setwd("/Users/abbou/Desktop/ter")


coupé<-fread("table_NA.csv",data.table=FALSE)
coupédataframe<-coupé   #to be able to do all the rest of my code without modifying dataframe in data
coupédataframe <- data.frame(coupé)
case<-sample(2,nrow(coupédataframe),replace=T)

#we separate our data set in 2
coupépartie1 <- coupédataframe[case==1,]
coupépartie2 <- coupédataframe[case==2,]

#then we separate each group in 20 so that everyone has an average of just under 50,000 observations
num<-sample(20,nrow(coupépartie1),replace=T)
coupédataframe1num1<-coupépartie1[num==1,]
coupédataframe1num2<-coupépartie1[num==2,]
coupédataframe1num3<-coupépartie1[num==3,]
coupédataframe1num4<-coupépartie1[num==4,]
coupédataframe1num5<-coupépartie1[num==5,]
coupédataframe1num6<-coupépartie1[num==6,]
coupédataframe1num7<-coupépartie1[num==7,]
coupédataframe1num8<-coupépartie1[num==8,]
coupédataframe1num9<-coupépartie1[num==9,]
coupédataframe1num10<-coupépartie1[num==10,]
coupédataframe1num11<-coupépartie1[num==11,]
coupédataframe1num12<-coupépartie1[num==12,]
coupédataframe1num13<-coupépartie1[num==13,]
coupédataframe1num14<-coupépartie1[num==14,]
coupédataframe1num15<-coupépartie1[num==15,]
coupédataframe1num16<-coupépartie1[num==16,]
coupédataframe1num17<-coupépartie1[num==17,]
coupédataframe1num18<-coupépartie1[num==18,]
coupédataframe1num19<-coupépartie1[num==19,]
coupédataframe1num20<-coupépartie1[num==20,]

coupédataframe1num1Knn<-kNN(coupédataframe1num1,k=5)
coupédataframe1num2Knn<-kNN(coupédataframe1num2)
coupédataframe1num3Knn<-kNN(coupédataframe1num3)
coupédataframe1num4Knn<-kNN(coupédataframe1num4)
coupédataframe1num5Knn<-kNN(coupédataframe1num5)
```

```
coupédataframe1num6Knn<-kNN(coupédataframe1num6)
coupédataframe1num7Knn<-kNN(coupédataframe1num7)
coupédataframe1num8Knn<-kNN(coupédataframe1num8)
coupédataframe1num9Knn<-kNN(coupédataframe1num9)
coupédataframe1num10Knn<-kNN(coupédataframe1num10)
coupédataframe1num11Knn<-kNN(coupédataframe1num11)
coupédataframe1num12Knn<-kNN(coupédataframe1num12)
coupédataframe1num13Knn<-kNN(coupédataframe1num13)
coupédataframe1num14Knn<-kNN(coupédataframe1num14)
coupédataframe1num15Knn<-kNN(coupédataframe1num15)
coupédataframe1num16Knn<-kNN(coupédataframe1num16)
coupédataframe1num17Knn<-kNN(coupédataframe1num17)
coupédataframe1num18Knn<-kNN(coupédataframe1num18)
coupédataframe1num19Knn<-kNN(coupédataframe1num19)
coupédataframe1num20Knn<-kNN(coupédataframe1num20)

coupédataframe1numKnnfusion =
rbind(coupédataframe1num1Knn,coupédataframe1num2Knn,coupédataframe1num3Knn,coupédataframe1num4Knn,coupédataframe1num5Knn,cou

View(coupédataframe1numKnnfusion)

num<-sample(20,nrow(coupépartie2),replace=T)
coupédataframe2num1<-coupépartie2[num==1,]
coupédataframe2num2<-coupépartie2[num==2,]
coupédataframe2num3<-coupépartie2[num==3,]
coupédataframe2num4<-coupépartie2[num==4,]
coupédataframe2num5<-coupépartie2[num==5,]
coupédataframe2num6<-coupépartie2[num==6,]
coupédataframe2num7<-coupépartie2[num==7,]
coupédataframe2num8<-coupépartie2[num==8,]
coupédataframe2num9<-coupépartie2[num==9,]
coupédataframe2num10<-coupépartie2[num==10,]
coupédataframe2num11<-coupépartie2[num==11,]
coupédataframe2num12<-coupépartie2[num==12,]
coupédataframe2num13<-coupépartie2[num==13,]
coupédataframe2num14<-coupépartie2[num==14,]
coupédataframe2num15<-coupépartie2[num==15,]
coupédataframe2num16<-coupépartie2[num==16,]
coupédataframe2num17<-coupépartie2[num==17,]
coupédataframe2num18<-coupépartie2[num==18,]
coupédataframe2num19<-coupépartie2[num==19,]
coupédataframe2num20<-coupépartie2[num==20,]

coupédataframe2num1Knn<-kNN(coupédataframe2num1)
coupédataframe2num2Knn<-kNN(coupédataframe2num2)
coupédataframe2num3Knn<-kNN(coupédataframe2num3)
coupédataframe2num4Knn<-kNN(coupédataframe2num4)
coupédataframe2num5Knn<-kNN(coupédataframe2num5)
coupédataframe2num6Knn<-kNN(coupédataframe2num6)
coupédataframe2num7Knn<-kNN(coupédataframe2num7)
coupédataframe2num8Knn<-kNN(coupédataframe2num8)
coupédataframe2num9Knn<-kNN(coupédataframe2num9)
coupédataframe2num10Knn<-kNN(coupédataframe2num10)
coupédataframe2num11Knn<-kNN(coupédataframe2num11)
coupédataframe2num12Knn<-kNN(coupédataframe2num12)
coupédataframe2num13Knn<-kNN(coupédataframe2num13)
coupédataframe2num14Knn<-kNN(coupédataframe2num14)
coupédataframe2num15Knn<-kNN(coupédataframe2num15)
coupédataframe2num16Knn<-kNN(coupédataframe2num16)
coupédataframe2num17Knn<-kNN(coupédataframe2num17)
coupédataframe2num18Knn<-kNN(coupédataframe2num18)
coupédataframe2num19Knn<-kNN(coupédataframe2num19)
coupédataframe2num20Knn<-kNN(coupédataframe2num20)

coupédataframe2numKnnfusion =
rbind(coupédataframe2num1Knn,coupédataframe2num2Knn,coupédataframe2num3Knn,coupédataframe2num4Knn,coupédataframe2num5Knn,cou

View(coupédataframe2numKnnfusion)
```

```
coupédataframeKnnfusion = rbind(coupédataframe2numKnnfusion, coupédataframe1numKnnfusion)

#calculate the percentage of missing values
 coupletavecvm<-read.csv("table_NA.csv", sep=",")
apply(coupletavecvm,2,function(x) sum(is.na(x)))->NbNAs #we count the number of missing values
nrow(coupletavecvm)->n #n = nombre de ligne
rbind(NbNAs,round(100*NbNAs/n,2))

#analysis of data
head(completavecvm)

str(completavecvm)

library(mice)
md.pattern(completavecvm)

#graphics

library(VIM)
aggr(completavecvm, col=c('navyblue','red'), numbers=TRUE, sortVars=TRUE, labels=names(data), cex.axis=.7,
gap=3, ylab=c("Histogram of missing data","Pattern"))

install.packages("Amelia")
library(Amelia)
missmap(completavecvm, main = "Missing values vs observed") #Display a graph with missing values vs
observed values
```

Code: Data Viz (Python, R)

```
#Importation of the needed libraries on Python


from plotly.offline import init_notebook_mode, iplot
import plotly.graph_objs as go
import numpy as np
%matplotlib inline
import plotly.plotly as py
from plotly import tools
import pandas as pd
import string, os, random
import calendar
import plotly
import plotly.offline as plt
import plotly.figure_factory as ff
import seaborn as sns
import matplotlib as mplt
import matplotlib.pyplot as plt

plotly.offline.init_notebook_mode(connected=True)

init_notebook_mode(connected=True)
punc = string.punctuation

#Barplot: Evolution of the number of road accidents in France

def create_stack_bar_data(col, df):
    aggregated = df[col].value_counts().sort_index()
    x_values = aggregated.index.tolist()
    y_values = aggregated.values.tolist()
    return x_values, y_values

x1, y1 = create_stack_bar_data('an', tabComp) #renvoie deux liste x1[les années] et y1 [les accidents ]
```

```
for i in range(len(x1)):
    x1[i] += 2000

trace1 = go.Bar(x=x1, y=y1, opacity=0.7, name="year count",marker =
dict(color=['midnightblue','midnightblue','midnightblue','midnightblue','midnightblue',
'midnightblue','midnightblue','midnightblue','darkgoldenrod','midnightblue','midnightblue',
'midnightblue']))
layout = dict(height=400, title='Evolution of the number of accidents in France',
legend=dict(orientation="h"), xaxis = dict(title = 'Year'), yaxis = dict(title = 'Number of Accidents'))
fig = go.Figure(data=[trace1], layout=layout)
iplot(fig)

#Data related to the localisation of accidents
#Importation of the needed libraries on R

library(flexdashboard)
library(knitr)
install.packages('DT')
library(DT)
install.packages('rpivotTable')
library(rpivotTable)
library(ggplot2)
install.packages('plotly')
library(plotly)
library(dplyr)
install.packages('openintro')
library(openintro)
install.packages('highcharter')
library(highcharter)
install.packages('maps')
library('maps')
install.packages("leaflet")
library(leaflet)
library(ade4)

## hcmap : Map showing the number of accidents according to French departments (R)

tablebynumacc<-group_by(tableComp,'num_acc')
car <- tablebynumacc %>%
       dplyr::group_by(dep) %>%
       dplyr::summarise(total= n())

totobon<-c(7835,4059,4562,4948,2383,4605,23838,2985,3360,4585,3453,9948,28447,12961,1753,5370,3838,52188,2261,10338,2669,501

mapdata <- get_data_from_map(download_map_data("countries/fr/fr-all-all"))
glimpse(mapdata)

set.seed(1234)

data_fake <- mapdata %>%
  select(code = `hc-a2`) %>%
  mutate(value = totobon)

glimpse(data_fake)
hcmap("countries/fr/fr-all-all", data = data_fake,
      name = "liste Dep", value = "value", joinBy = c("hc-a2", "code"),
      dataLabels = list(enabled = TRUE, format = '{point.name}'),
       borderColor = "transparent",borderWidth = 0.5)

##Donut Chart : Accidents by type of roads

data_2 = tabComp.groupby(by='catr', as_index=False).count()
Title = " "
def PlotPiechart(labels, values, columnName):
    fig = {
      "data": [
        {
          "labels": labels,
```

```
          "values": values.num_acc,
          #"domain": {"x": [0, 1]},
          "name": columnName,
          "hoverinfo":"label+percent+name",
          "hole": .6,
          "type": "pie"
        },
         ],
      "layout": {
            "title":Title,
            "annotations": [
                {
                    "font": {
                        "size": 40
                    },
                    "showarrow": False,
                    "text": " ",
                    "x": 5.50,
                     "y": 0.5
                }
            ]
        }
    }
    plt.iplot(fig)

catrLabel = {1:'Highway',
                2:'National Road',
                3:'Departmental Road',
                4:'Communal Way',
                5:'Off-Public Network',
                6:'Parking Lot',
                7:'Other'}
data_2_sort= data_2.sort_values(by='num_acc')
data_2.catr = data_2.catr.map(catrLabel)
Title = "Percentage of accidents according to the type of road"
PlotPiechart(data_2.catr.values, data_2, 'CategoryOfRoad')

##Grouped Bar Chart : Severity of the accidents in and out an agglomeration (R)

dat<-data.frame(
agg=factor(c('Out of agglomeration','In built-up areas','Out of agglomeration','In built-up areas','Out of
agglomeration','In built-up areas')),
gravity=factor(c('unscathed','unscathed','slightly injured','slightly injured','hospitalized or
killed','hospitalized or killed'),levels=c('unscathed','slightly injured','hospitalized or killed')),
Number_of_accidents= c(226447,538427,180116,486757,222783,221475)
)

ggplot(dat,aes(x=agg,y=Number_of_accidents,fill=gravity))+geom_bar(stat="identity",position =
position_dodge())+ggtitle(label=' In wich areas do more serious accidents occur ?
')+scale_fill_manual(values = c("grey",'aquamarine4',"salmon"))

##Donut Chart : Accidents by type of intersection

data_6 = tabComp.groupby(by='inter', as_index=False).count()
Title = " "

interLabel = {1:'Out of intersection',
                2:'Intersection in X',
                3:'Intersection in T',
                4:'Intersection in Y',
                5:'Intersection with more than 4 branches',
                6:'Giratory',
                7:'Place',
                8:'Level crossing',
                9:'Other intersection'}

data_6_sort= data_6.sort_values(by='num_acc')
data_6.inter = data_6.inter.map(interLabel)
```

```
Title = "Percentage of accidents according to the type of intersection"
PlotPiechart(data_6.inter.values, data_6, 'CategoryOfIntersection')

#Data related to the weather conditions
##Donut Chart : Accidents according to the atmospheric conditions

data_7 = tabComp.groupby(by='atm', as_index=False).count()

atmLabel = {1:'Normal',
     2:'Light rain',
     3:'Heavy rain',
     4:'Snow - hail',
     5:'Fog - smoke',
     6:'Strong wind - storm',
     7:'Dazzling weather',
     8:' Cloudy weather',
     9:'Other'}
data_7_sort= data_7.sort_values(by='num_acc')
data_7.atm = data_7.atm.map(atmLabel)
Title = "Percentage of accidents according to meteorologic conditions"
PlotPiechart(data_7.atm.values, data_7,'meteorologicCond')

##stacked bar chart: Accidents according to lighting condition (sous R studio)

dat<-data.frame(
brightness=factor(c('Full day','Twilight or dawn', 'Night without public lighting','Night with public
lighting not lit','Night with public lighting on')),
gravity=factor(c('unscathed','unscathed','unscathed','unscathed','unscathed','slightly injured','slightly
injured','slightly injured','slightly injured','slightly injured','hospitalized or killed','hospitalized or
killed','hospitalized or killed','hospitalized or killed','hospitalized or
killed')),levels=c('unscathed','slightly injured','hospitalized or killed'),
Number_of_accidents=c(548487,42066,48183,5763,120375,460962,36784,40151,5583,123393,287330,28460,62557,3749,62162)
)

ggplot(dat,aes(x=brightness,y=Number_of_accidents,fill=gravity))+geom_bar(stat="identity",position='stack')+
ggtitle(label='')+scale_fill_manual(values = c( "pink2","dimgrey",   "lightblue" ))+
theme(axis.title.x = element_text(vjust = 0))

##Donut Chart : Accidents according to the surface of road condition

data_4 = tabComp.groupby(by='surf', as_index=False).count()

surfLabel = {1:'normal',
                2:'wet',
                3:'puddles',
                4:'flooded',
                5:'snow',
                6:'mud',
                7:'icy',
                8:'fat-oil',
                9:'other'}
data_4_sort= data_4.sort_values(by='num_acc')
data_4.surf = data_4.surf.map(surfLabel)
Title = "Percentage of accidents according to road surface condition"
PlotPiechart(data_4.surf.values, data_4,'surface condition')

#Data related to the date
##The number of accidents per month and hours

x1, y1 = create_stack_bar_data('mois', tabComp)
x1 = ['January', 'February', 'March', 'April', 'May', 'June',
'July','August','September','October','November','December']
trace1 = go.Bar(x=x1, y=y1, opacity=0.75, name="Month", marker=dict(color='olivedrab'))

x2, y2 = create_stack_bar_data('hrmn', tabComp)
trace2 = go.Bar(x = x2, y = y2, opacity = 0.75, marker=dict(color='salmon'), name = "Hours")

fig = tools.make_subplots(rows = 2, cols = 1)
```

```
fig.append_trace(trace1, 1, 1)
fig.append_trace(trace2, 2, 1)
layout = dict(height=900, title='Accidents by times');
fig.layout.update(layout)
#fig['layout'].update(height=800,title='Accidents by Type of Road')
iplot(fig, filename='stacked-bar')

##What our data can also teach us
#Violin plot : repartition of the age of people, according to their sex and the reasons of their journey

sns.violinplot(x="trajet", y="age",hue="sexe", data=tabComp, palette="YlOrBr",split=True)

## Donut Chart : Accidents per reason of travelling

data_8 = tabComp.groupby(by='trajet', as_index=False).count()

trajLabel = {1:'Home-Work',
     2:'Home-school',
     3:'Shopping-Shopping',
     4:'Professional use',
     5:'Promenade-leisure',
     9:'Other'}
data_8_sort= data_8.sort_values(by='num_acc')
data_8.trajet = data_8.trajet.map(trajLabel)
Title = "Percentage of accidents according to trajet"
trace1 =PlotPiechart(data_8.trajet.values, data_8,'trajet')

trace2=sns.violinplot(x="trajet", y="age",hue="sexe", data=tabComp, palette="Pastel2_r",split=True)


##Times séries visualization

#We change the form of the columns an, mois and jour: %Y-%m-%d

tabComp['an'] = tabComp.an + 2000
for name in ['an','mois','jour']:
    tabComp[name] = tabComp[name].astype('str')

tabComp['Date'] = tabComp.an + '-' + tabComp.mois + '-' + tabComp.jour
tabComp['Date'] = pd.to_datetime(tabComp.Date, format='%Y-%m-%d')

data_9 = tabComp.groupby(by='Date', as_index=False).count()
data_9 = data_9.sort_values(by='Date')

data_9_1 = go.Scatter(x=data_9.Date, y=data_9.num_acc, name = "Actual Number", line = dict(color =
'#17BECF'))
data_9_2 = go.Scatter(x=data_9.Date, y=data_9.num_acc.rolling(9, min_periods=1).mean(), name = "Rolling Avg
of Week", line = dict(color = '#FCEB71'))
data_9_3 = go.Scatter(x=data_9.Date, y=data_9.num_acc.rolling(9, min_periods=1).std(), name = "Rolling Std
of Week", line = dict(color = '#FC5B71'))
Title = 'Time line graph how much accident happened daily.'
layout = dict(
    title=Title,
    xaxis=dict(
        rangeselector=dict(
            buttons=list([
                dict(count=1,
                     label='1m',
                     step='month',
                     stepmode='backward'),
                dict(count=6,
                     label='6m',
                     step='month',
                     stepmode='backward'),
                dict(count=12,
                     label='1Y',
                     step='month',
```

```
                    stepmode='backward'),
                dict(count=24,
                    label='2Y',
                    step='month',
                    stepmode='backward'),
                dict(step='all')
            ]) ),rangeslider=dict(),type='date' )
data = go.Figure(data=[data_9_1, data_9_2,data_9_3], layout=layout)
plt.iplot(data)
```

Code: Econometric part data treatments (STATA, R)

```
#We delete all the observations that contain missing values
#We delete the variables \lat", \lon" and \dep" as we won't use them here
Table_coupee <-newtable[!is.na(newtable$trajet), c(1:21,25:35)]
for (i in 1:32){
  Table_coupee <-Table_coupee[!is.na(Table_coupee[,i]),]
}
#We change the id
Table_coupee$id = seq(1:829591)
#We export the cut table as "Table_sansNA.csv"
write.csv(Table_coupee, "Table_sansNA.csv")

###We finish cleansing our table on STATA
import delimited "C:\Users\saraa\OneDrive\Bureau\Cours_ISFA\TER\table_sansNA.csv"

#We delete voie and v1(useless), then we delete an_nais and an
Drop an_nais an voie v1

###We exported the new table as "Table_prete_stata.csv"
export delimited using "C:\Users\saraa\OneDrive\Bureau\Cours_ISFA\TER\Table_prete_stata.csv", replace

#We reorganize the modalities of our variables
T <- read.csv("Table_prete_stata.csv")
T<-T[which(T$place != 0),] #We delete the pedestrians
T$place <- replace(T$place, which(T$place!=1),0) #place = 1 if driver, else 0
T$sexe <- replace(T$sexe, which(T$sexe == 2), 0) #sexe = 1 if homme, else 0
T$trajet <- replace(T$trajet, which(T$trajet == 2), 1) # trajet = 1 if pro
T$trajet <- replace(T$trajet, which(T$trajet == 4), 1)
T$trajet <- replace(T$trajet, which(T$trajet == 3),0) # trajet = 0 if loisirs
T$trajet <- replace(T$trajet, which(T$trajet == 5),0)
T$trajet <- replace(T$trajet, which(T$trajet == 9), 2) #trajet = 2 if other reasons
T$mois <- replace(T$mois, which(T$mois == 12), 0) #mois = 0 if winter
T$mois <- replace(T$mois, which(T$mois == 1), 0)
T$mois <- replace(T$mois, which(T$mois == 2), 0)
T$mois <- replace(T$mois, which(T$mois == 3), 1) #mois = 1 if spring
T$mois <- replace(T$mois, which(T$mois == 4), 1)
T$mois <- replace(T$mois, which(T$mois == 5), 1)
T$mois <- replace(T$mois, which(T$mois == 6), 2) #mois = 2 if summer
T$mois <- replace(T$mois, which(T$mois == 7), 2)
T$mois <- replace(T$mois, which(T$mois == 8), 2)
T$mois <- replace(T$mois, which(T$mois == 9), 3) # mois = 3 if autumn
T$mois <- replace(T$mois, which(T$mois == 10), 3)
T$mois <- replace(T$mois, which(T$mois == 11), 3)
T$atm <- replace(T$atm, which(T$atm == 8),1) # atm = 1 if normal weather
T$atm <- replace(T$atm, which(!(T$atm == 1)),0) #atm = 0 if bad weather
icol <-which(T$col !=7) #indices of the table where there were a collision
coli = rep(0, 829575)
coli[icol] <- 1
T<-cbind(T,coli) #We add a variable "coli" : 1 if there were a collision, else 0
T$col <- replace(T$col, which(T$col*T$coli == 0), 0) #Col = 0 if no collision, x if collision of type x
T$hrmn <- replace(T$hrmn, which(T$hrmn == 1),0) #hrmn = 1 if peak hour, else 0
T$hrmn <- replace(T$hrmn, which(T$hrmn >= 630 & T$hrmn <=900),1)
T$hrmn <- replace(T$hrmn, which(T$hrmn >= 1600 & T$hrmn <=1900),1)
```

```
T$hrmn <- replace(T$hrmn, which(!(T$hrmn == 1)),0)
T$agg <- replace(T$agg, which(T$agg == 1), 0) #agg = 1 if in an agglomeration, else 0
T$agg <- replace(T$agg, which(T$agg == 2), 1)
T$lum <- replace(T$lum, which(T$lum == 3), 0) #lum = 0 if dark night, 1 if day light, 2 si morning light, 3
si lightened night
T$lum <- replace(T$lum, which(T$lum == 4), 0)
T$lum <- replace(T$lum, which(T$lum == 5), 3)
T$catr <- replace(T$catr, which(T$catr == 4), 0) #catr = 0 if other, 1 if highway, 2 if national road, 3
departemental road
T$catr <- replace(T$catr, which(T$catr == 5), 0)
T$catr <- replace(T$catr, which(T$catr == 6), 0)
T$catr <- replace(T$catr, which(T$catr == 9), 0)
T$plan <- replace(T$plan, which(T$plan == 1), 0) # plan = 1 if turn, else 0
T$plan <- replace(T$plan, which(T$plan == 2), 1)
T$plan <- replace(T$plan, which(T$plan == 3), 1)
T$plan <- replace(T$plan, which(T$plan == 4), 1)
i_inters <- which(T$inter != 1)
inters <- rep(0,829575)
inters[i_inters] <- 1
T <- cbind(T, inters) #we add a variable "inters" = 1 if intersection , else 0
T$inter <- replace(T$inter, which(T$inter*T$inters == 0), 0)
T$inter <- replace(T$inter, which(T$inter == 4), 1) #inter = 0  if no intersection, 1 if other, 2 if X, 3
if T
T$inter <- replace(T$inter, which(T$inter == 5), 1)
T$inter <- replace(T$inter, which(T$inter == 6), 1)
T$inter <- replace(T$inter, which(T$inter == 7), 1)
T$inter <- replace(T$inter, which(T$inter == 8), 1)
T$inter <- replace(T$inter, which(T$inter == 9), 1)
T$circ <- replace(T$circ, which(T$circ == 3), 0) #circ = 0 if other, 1 if one way, 2 if 2-way road
T$circ <- replace(T$circ, which(T$circ == 4), 0)
T$prof <- replace(T$prof, which(T$prof == 2), 0) # prof = 1 if flat road, else 0
T$prof <- replace(T$prof, which(T$prof == 3), 0)
T$prof <- replace(T$prof, which(T$prof == 4), 0)
T$catv <- replace(T$catv, which((T$catv !=2 & T$catv != 7 & T$catv != 10 & T$catv != 33 )),0) #catv = 0 if
other, 2,7,10 or 33 if type 2,7,10 or 33
T$surf <- replace(T$surf, which(T$surf !=1), 2) #surf = 1 if slippery road, else 0
T$surf <- replace(T$surf, which(T$surf ==1), 0)
T$surf <- replace(T$surf, which(T$surf ==2), 1)
T$secu <- replace(T$secu, which(T$secu%%10 == 1),1) #Secu = 1 if a security system was present, else 0
T$secu <- replace(T$secu, which(T$secu%%10 == 2),0)

#We export the new table as \Table_Eco.csv"
write.csv(T,"Table_Eco.csv")



#We rename the variables that became dichotomies
import delimited "C:\Users\saraa\OneDrive\Bureau\Cours_ISFA\TER\Table_Eco.csv"
rename place driver
rename sexe man
rename atm normal_wether
rename hrmn peak_hour
rename plan turn
rename prof flat_road
rename surf slippery
drop v1 #We have a new variable R created, but we don't want it

#We create the other dichotomic variables
tab trajet, gen(reason)
rename reason1 loisirs_reason
rename reason2 pro_reason
rename reason3 other_reason
tab mois, gen(s)
rename s1 winter
rename s2 spring
rename s3 summer
rename s4 autumn
```

```
tab lum, gen(l)
rename l1 dark_night
rename l2 day_light
rename l3 morn_light
rename l4 lightened_night
tab inter, gen(i)
rename i1 intersection
rename i2 other_inter
rename i3 X_inter
rename i4 T_inter
drop inters
tab col, gen(c)
drop c1
rename c2 face_2_col
rename c3 file_2_col
rename c4 cote_2_col
rename c5 file_plus_col
rename c6 plus_multi_col
rename c7 autres_col
tab catr, gen(c)
rename c1 other_road
rename c2 highway
rename c3 nat_road
rename c4 dep_road
tab circ, gen(c)
rename c1 other_way_road
rename c2 one_way_road
rename c3 two_way_road
tab catv, gen(v)
rename v1 other_veh
rename v2 veh_type2
rename v3 veh_type7
rename v4 veh_type10
rename v5 veh_type33

#We delete the variables we transformed into dichotomies
drop trajet mois lum inter col catr circ catv

#We export the FINALE table as \Table_finale_eco.csv"
export delimited using "C:\Users\saraa\OneDrive\Bureau\Cours_ISFA\TER\Table_finale_eco.csv", replace

#Predicted probabilities for people to be unscathed, slightly or seriously injured :
mfx, predict(outcome(3))
mfx, predict(outcome(2))
mfx, predict(outcome(1))

#predicted probabilities to be seriously injured with and without a collision:
mfx, predict(outcome(3)) at (coli=0)
mfx, predict(outcome(3)) at (coli=1)

#predicted probabilities by vehicle type:
forvalues i=1/3 {
mfx, predict(outcome(‘i’)) at(face_2_col =1 file_2_col =0 cote_2_col=0 file_plus_col=0 plus_multi_col=0
autres_col=0 coli=1)
mfx, predict(outcome(‘i’)) at(face_2_col =0 file_2_col =1 cote_2_col=0 file_plus_col=0 plus_multi_col=0
autres_col=0 coli=1)
mfx, predict(outcome(‘i’)) at(face_2_col =0 file_2_col =0 cote_2_col=1 file_plus_col=0 plus_multi_col=0
autres_col=0 coli=1)
mfx, predict(outcome(‘i’)) at(face_2_col =0 file_2_col =0 cote_2_col=0 file_plus_col=1 plus_multi_col=0
autres_col=0 coli=1)
mfx, predict(outcome(‘i’)) at(face_2_col =0 file_2_col =0 cote_2_col=0 file_plus_col=0 plus_multi_col=1
autres_col=0 coli=1)
mfx, predict(outcome(‘i’)) at(face_2_col =0 file_2_col =0 cote_2_col=0 file_plus_col=0 plus_multi_col=0
autres_col=1 coli=1)
}

#Predicted probabilities with and without using a security system
mfx, predict(outcome(3)) at (secu=0)
```

```
mfx, predict(outcome(3)) at (secu=1)
mfx, predict(outcome(2)) at (secu=0)
mfx, predict(outcome(2)) at (secu=1)
mfx, predict(outcome(1)) at (secu=0)
mfx, predict(outcome(1)) at (secu=1)

#We calculate the maximum number of passengers on vehicles of type 33:

data<- read.csv("Table_finale_eco.csv", sep=",", header=TRUE)
data<-data[which(data$veh_type33==1),]
data <-data[order(data$num_acc),]

max = 0
j=0
for (i in 1:length(data)-1){
  j=1
  while (i+j<=62559 && data$driver[i] == 0 && data$driver[i+j] == 0 && (data$num_acc[i]==data$num_acc[i+1])
){
    j=j+1
  }
  if (j>max){
    max=j
  }
}
max
> max
[1] 1

#We calculate the number of correct predictions:
predict p1-p3
#We check they are probabilities:
gen p = p1+p2+p3
#That's ok as we get 1 everywhere
gen grvpred=1 if p1== max(p1,p2,p3)
replace grvpred=2 if p2== max(p1,p2,p3)
replace grvpred=3 if p3== max(p1,p2,p3)
gen nb11=1 if hltpred==grav & grav=1
gen nb22=1 if hltpred==grav & grav=2
gen nb33=1 if hltpred==grav & grav=3
gen nb12=1 if hltpred==1 & grav=2
gen nb13=1 if hltpred==1 & grav=3
gen nb21=1 if hltpred==2 & grav=1
gen nb23=1 if hltpred==2 & grav=3
gen nb31=1 if hltpred==3 & grav=1
gen nb32=1 if hltpred==3 & grav=2
```

6.3 Sources

- https://www.preventionroutiere.asso.fr/2016/04/22/statistiques-daccidents/
- https://www.kaggle.com/ahmedlahlou/accidents-in-france-from-2005-to-2016#places.csv
- https://datascienceplus.com/imputing-missing-data-with-r-mice-package/
- https://www.math.univ-toulouse.fr/~besse/Wikistat/pdf/st-m-app-idm.pdf
- https://mrmint.fr/donnees-manquantes-data-science
- https://statistics.ohlsen-web.de/multiple-imputation-with-mice/
- https://blog.affini-tech.com/r-package-doparallel/
- https://cran.r-project.org/web/packages/naniar/readme/README.html
- http://cs.joensuu.fi/~villeh/icpr2004.pdf
- https://rstudiopubsstatic.s3.amazonaws.com/316172_a857ca788d1441f8be1bcd1e31f0e875.html
- https://rstudio-pubs-static.s3.amazonaws.com/316172_a857ca788d1441f8be1bcd1e31f0e875.html
- https://www.r-bloggers.com/identify-describe-plot-and-remove-the-outliers-from-the-dat
- Ouvrage Statistical Analysis with Missing Data, Little et Rubin
  Journal de la Société Francaise de Statistique Vol. 159 No. 2 (2018)
- Exploring, handling, imputing and evaluating missing data in statistical analyses: a review ofexisting approaches
  Alyssa Imbert1 et Nathalie Vialaneix1
- http://www.securite-routiere.gouv.fr/medias/espace-presse/publications-presse/bilan-definitif-de-l-accidentalite-routiere-2013

- https://www.vie-publique.fr/politiques-publiques/politique-route-securite-routiere/securite-routiere/
- http://www.fiches-auto.fr/articles-auto/l-auto-en-chiffres/s-582-statistiques-sur-les-accidents-de-le-route-mortalite-contexte-etc.php
- http://temis.documentation.developpement-durable.gouv.fr/pj/15062/15062_3.pdf
- https://www.turbo.fr/actualite-automobile/bmw-un-systeme-daide-la-conduite-jamais-vu-video-41952
- http://www.lefigaro.fr/social/2017/09/26/20011-20170926ARTFIG00074-securite-routiere-les-trajets-professionnels-sont-les-plus-exposes-au-risque-d-accidents.php
- http://www.numdam.org/article/RSA_1960__8_4_15_0.pdf
- https://www.analyticsvidhya.com/blog/2016/02/time-series-forecasting-codes-python/
- https://www.academia.edu/15346300/conomtrie_applique_avec_Stata
- http://www.univ-orleans.fr/deg/masters/ESA/CH/Qualitatif_Chapitre2.pdf
- https://stats.idre.ucla.edu/stata/dae/ordered-logistic-regression/
- http://coursexercices.com/getfile.php?file=https://abenkhalifa.files.wordpress.com/2015/02/formation_stata.pdf&title=Visitez__CoursExercices.com____formation_stata.pdf_285
- https://dss.princeton.edu/training/Margins.pdf
- Nathalie Havet's lessons