Data Science Project

Supervised by:

Mme Dorra Trabelsi
Mr Med Anis Ben Lasmar
Mme Ines Mhaya

Elaborated by:

Fairouz Arfaoui
Hosni Ayachi
Mohamed Malek Douik
Nizar Masmoudi
Sarra Ben Abdessalem
Skander Marnissi

4DS2

2019-2020

Table of contents:

Introduction

As the Vice President of risk management for Renaissance reinsurance Mr. JD LONG said: “*The future of insurance companies is of we can get the actuaries and the data scientists talking together about what are the strengths and weaknesses of our different methodologies.*” Indeed, machine learning is becoming more and more essential in the insurance industry given its many applications such as insurance advice, customer retention and risk management.

In this context, we were asked by the CGA (‘Commité Générale des Assurances’ or General Committee of Insurances Compagnies) to use the data source provided in order to help it classify clients using Bonus-Malus indicator.

All along this report we are going to present our solution for the risk management fraud detection and contract termination uses cases.

CHAPTER I:

Data Presentation

1.1 Data source: identification and description:

- **1.1.1 Internal data:**

This dataset holds a large amount of information collected by the CGA from all the insurance companies in order to track the accident history of every client and classify them in different categories using the BonusMalus indicator. It consists of the following tables:

-**Bonus-Malus**: Represents the Bonus-Malus for each client which is an indicator calculated by insurance companies depending on the number of cases of claims they had.

-**Epave**: Contains mainly the ruins of clients' vehicles in extreme cases

-**Marque**: It contains the brands of the clients' vehicles

-**Police**: It contains the contracts between the clients and the insurance agencies

-**Sinistre**: Contains different cases of claims

-**Usage**: Represents the usage of vehicles (for example: taxis, public transport etc)

-**Véhicule**: Represents all the insured vehicles

-**Assuré**: It contains the clients' information, naming the id and the date of their license obtaining

- **1.1.2 External data**:

ONSR(Observatoire National de la Sécurité Routière) Website: We scraped the ONSR website for car accidents statistics from the year 2016 to the present year

Questionnaire: We used Google Forms to create a form containing basic questions for insured drivers as well as a personality test for drivers. We sent it to a mail list we scraped from the frontend of an unsecured website.

Web Scraping:

-Interior affairs ministry: with the library Beautiful Soup

-Site Edmunds It's a cars brands and models review web site.We used the library Selenium.

-Reddit : with Praw. We declared an instance Reddit developer with which we accessed the subreddits of list of cars we predefined. For each thread we scrapped the comments that we sentiment-analysed with Textblob.

-Twitter : We prefered using Twitterscrapper rather than tweepy because not only doesn't require a developer account, but it also allows us to run research request with the space bar.

1.2 Business objectives:

The main objective of this project is to help the CGA gain a better perspective not only about the types of clients insured but also about the insurance agencies activity. The purpose of this is to reveal the specific reasons behind the alarming number of car incidents in Tunisia. Therefore, our aim will be to develop an app capable of preventing risky relations with certain clients based on a scoring operation.

Since such kind of app are already being launched in the market (by MITIGAN for example), we will also work on fraud detection, either by clients or agencies, as well as on contract termination prediction.

1.3 Data Science objectives:

The business and the technical objectives of this projects go hand on hand as the former cannot be accomplished without the latter. Accordingly, we will be using the following different techniques:

-Featuring engineering: This is the first step. It consists of cleaning the data and maneuvering it in order to giving it synthesized meaning and purpose. This is a primordial measure for the following steps.

-Applying big data procedures:

* Web scrapping: Collecting information from the web regarding our data using

* Data storing: Using a NoSQL database (Cassandra in our case) to store our data (internal and external)

* Clustering: Using HDFS (Hadoop Distributed File System) to distribute our data on different machines and use Spark for the treatment.

- Machine Learning: Applying different machine learning models (supervised and unsupervised) to classify our data and extracting distinguished profiles and behaviors. This step will help us not only score the clients, the vehicles and/or the agency but also detect fraud and contract termination cases (if they exist).

CHAPTER II:

Data Preparation

2.1 Internal data preparation:

2.1.1 Raw separate tables:

We ran the following operations on all the tables we received from the CGA

Operation Table	Eliminating Columns	Creating new columns	Removing missing values /duplicates	Data Conversion
Assuré		'Gouvernorat' generated by 'CodePostal'	We dropped duplicate Ids	.'dateObtention' : to_datetime .'Code_postal' : to_int
BonusMalus	7 out of 26 : 3 empty 4 useless			3 out of 19 were changed to datatime
Marque	6 out of 8 are useless columns			
Police	4 out of 17 3 empty 1 useless	'resiliation' and 'suspension'		3 out of 13 were changed to datetime
Sinistre	3 out of 21 1 empty 2 useless		3 out of 18 were affected	
Usage	3 out of 5 3 useless			
Véhicule	5 out of 15 5 empty		2 out of 10 columns were affected	3 out of 10 columns were changed

BonusMalus:

Elimination empty columns: ['consulter', 'codeGouvernorat', 'statut']

Eliminating useless columns: ['classeBonusMalusCompagnie', 'classeBonusMalusCGA', 'coefBonusMalusCompagnie', 'coefBonusMalusCGA']

DataConversion: ['consulter', 'codeGouvernorat', 'statut']

Marque:

Eliminating useless columns: ['DESIGA', 'OBSERV', 'DDEB', 'DFIN', 'CD_MARQ_CH', 'CD_CONST']

Police:

Eliminating empty columns: ['codeCourtierCGA', 'resiliationEcheance', 'dateRemiseEnVigueure']

Eliminating useless columns: ['verrouillageModifPolice']

Data_Conversion: ['dateEffetPolice', 'dateExpirationPolice', 'dateSuspension']

Creating new columns: Since ‘dateResilation’ and ‘datesuspension’ have a lot of missing values and that they just hold the information that the client is either suspended or quit the insurance company, we decided to replace it by a binary variable that we will use in the termination contract cases objectives.

Sinistre:

Elimination empty columns: ['date']

Eliminating useless columns: ['sinistre_id', 'mouvementDusinistre']

Elimination missing values: ['numeroImmatriculationVehiculeAdverse', 'typeImmatriculationVehiculeAdverse', 'codeCompagnieAdverse']

Usage:

Eliminating useless columns: ['id', 'ENABLED', 'CODE_STR']

Véhicule:

Elimination empty columns: ['dateDerniereVisite', 'dateMiseEpave', 'dateRetrait', 'dateMiseCirculation', 'dateMiseAJourVehicule']

Data conversion: ['dateInsertion', 'dateAjout'] were changed to datetime

['puissanceFiscal'] was changed to int

Elimination missing values: [‘energie’, ‘puissanceFiscale’]

This UML diagram summarizes the data left:

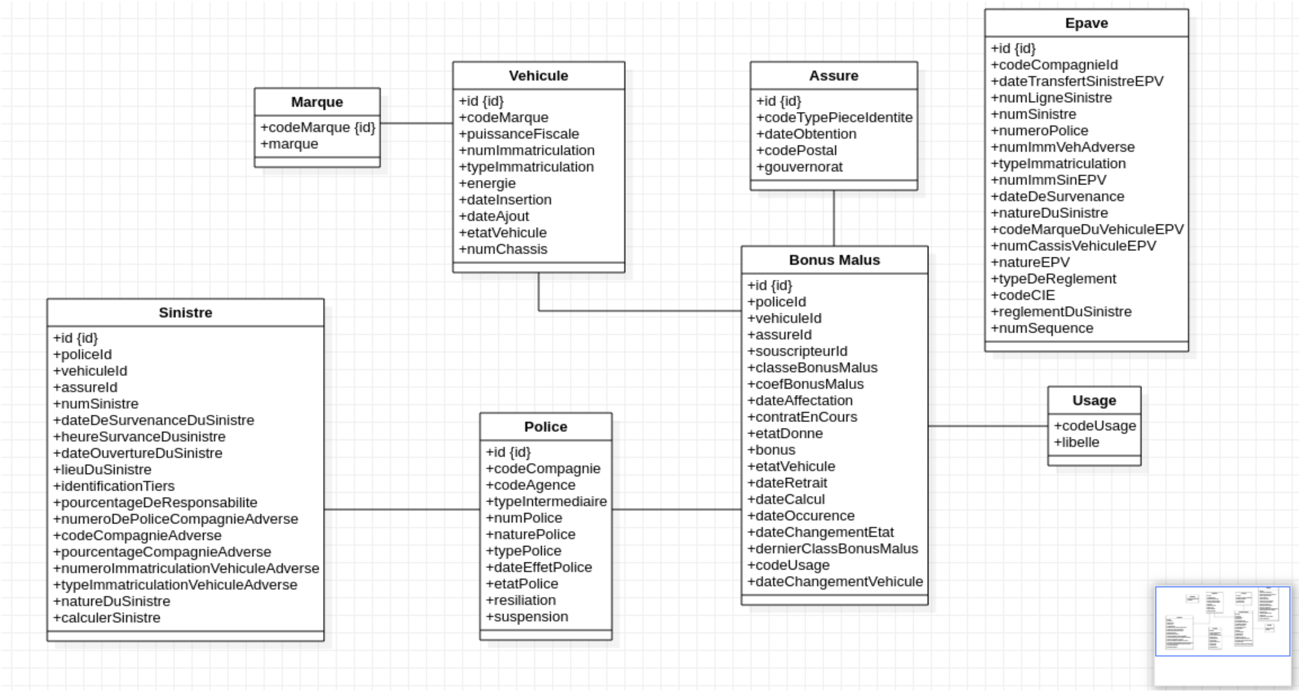


Figure 1 Merged tables

After merging our initial tables we wound up having around 33 000 lines from 90 000 which represents around 1/3 of the information given.

The different reasons for eliminating certain columns are summed up as following:

REASON	Correlation with another existing variable	Lack of use for our business objectives	Constant values of the modalities	Foreign keys
COLUMNS	'naturePolice', 'typePolice', 'bonus','resiliation'	'numChassis', 'dernierClassBonusMAllus', 'codeTypePieceIdentite', 'calculerSinistre', 'typeIntermediaire', 'dateInsertion', 'dateAjout', 'dateAffectation', 'codePostal', 'numeroDePoliceCompagnieAdverse',	'suspension', 'etatVehicule',	'vehiculeId', 'codeUsage', 'sinistreId', 'numPolice', 'numSinistre', 'codeMarque',

2.1.2 Data Comprehension:

Internal Data:

Let's start by casting the table 'Epave' away since it represents vehicles that can't circulate anymore and that it wasn't useful for our fixed objectives.

As for the rest of our internal data when we examined it we came across a few anomalies. The first one is that more than 20 000 have the modality '0 in the variable 'contratEnCours' while they are marked as 'V' in the variable 'etatPolice' meaning 'en vigueur' or contract in effect.

The second one lies in an abnormal behavior detected in a few clients who have an unusually big number of vehicles. For example the client whose id is '53968' has more than 400 vehicles, while it should have been saved as 'fleet' it's actually 'individual'.

The third anomaly concerns registered vehicles and the number of contracts, if we dig deeper we'll see that more than 400 vehicles have two contracts at the same time which is illegal.

These anomalies push us to question whether it's an honest mistake of typing or if it's actually fraud.

External Data:

From the scraping of both Twitter and Reddit we obtained the following table of cars scoring:

Variable	Type
Id	Int
Car	Varchar
Comment-tweet	Varchar
From	Varchar
Hashtags	List
Polarity	Float
Polarity_meaning	varchar

In order to only keep the useless variable we ended up having a table containing the Id, Car and Score. From this new table we visualized the top 10 first best makes of cars (less included in accidents).

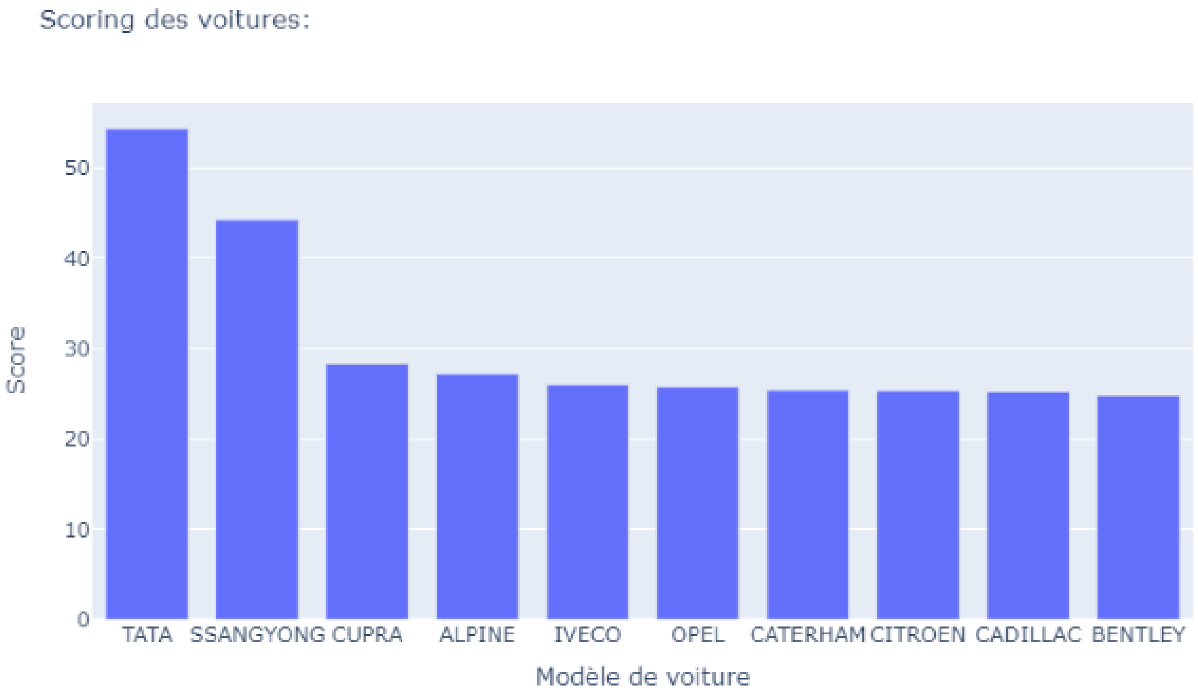


Figure 2 : Vehicle makes scoring

ONSR Website: This data contains accidents that happened in Tunisia from 2016 to 2020. We merged all the data frames related to each year and added a column containing the year corresponding. We also deleted the percentage and the total variables because we can deduct it from the numbers we have**.** We then generated this chart showing the number of accidents per month and the number of accidents per region. In coherence with our internal data, Tunis seems to be the governorate in which most accidents take place.

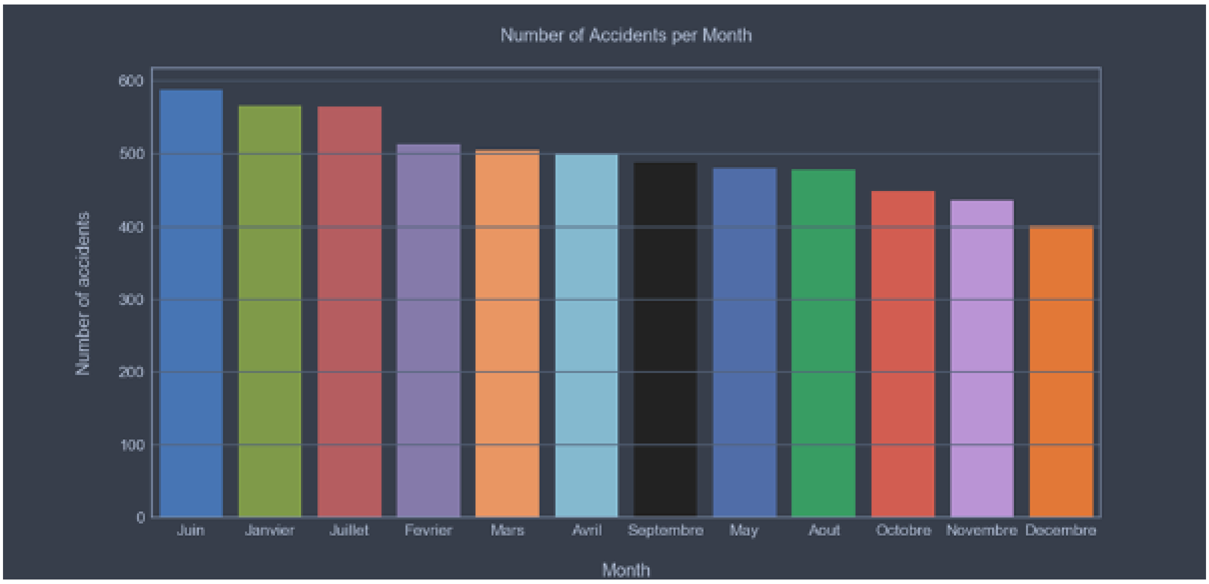


Figure 3: Number of accidents per month

Comment: June is in lead of the months when most accidents happen, and December is the last. A possible explanation is that June is the first month of summer holidays so more and more people go out while during winter holidays it is more common to enjoy them inside.

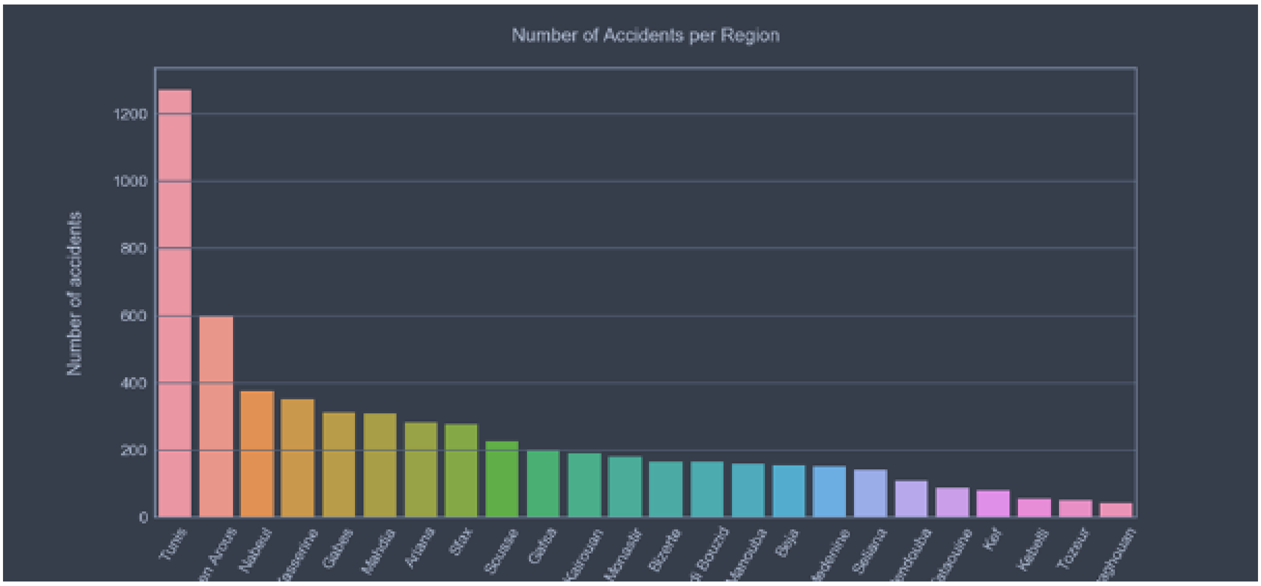


Figure 4: Number of accidents per region

Comment: Tunis is in the lead while Zaghouan is the last. It could be explained by the fact that the capital has the highest population number and that it has more data saved.

Questionnaire: We retrieved all the answers we received for the form in a CSV file. Since the questionnaire is controlled for obligatory questions we didn't have any null values.

The category variables that represent the personality test were replaced by scores given for each answer of the questions of the said test. A global score was calculated according to certain coefficients that reflect the importance of the question. In consequence we generated a new column that represents the score of the driver which replaced all 10 columns of the personality test.

However, we found at the end that the results are biased because the sample isn't representative enough.

2.3 Feature selection:

2.3.1 Client Classification:

In this section We decided to work on two main axes, Bonus-Malus and Sinistre. Each axis is associated to 3 tables (Police, Vehicule and Assuré) through foreign keys.

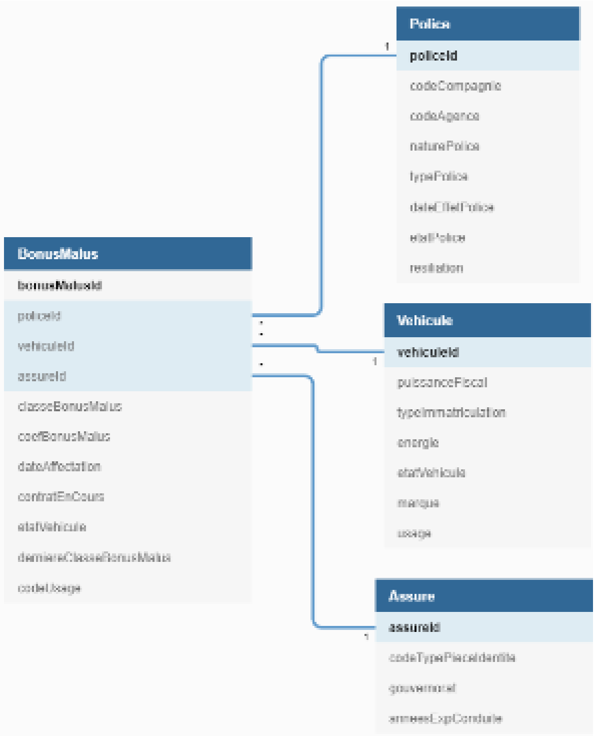


Figure 5: Bonus Malus relations

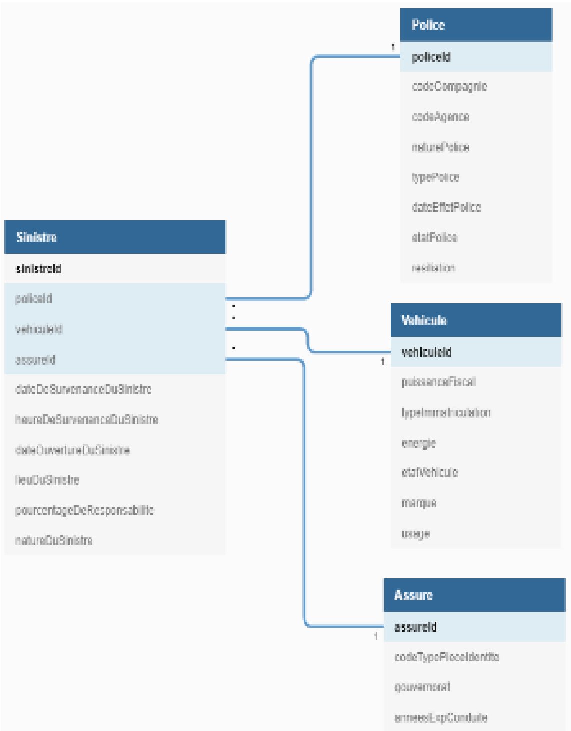
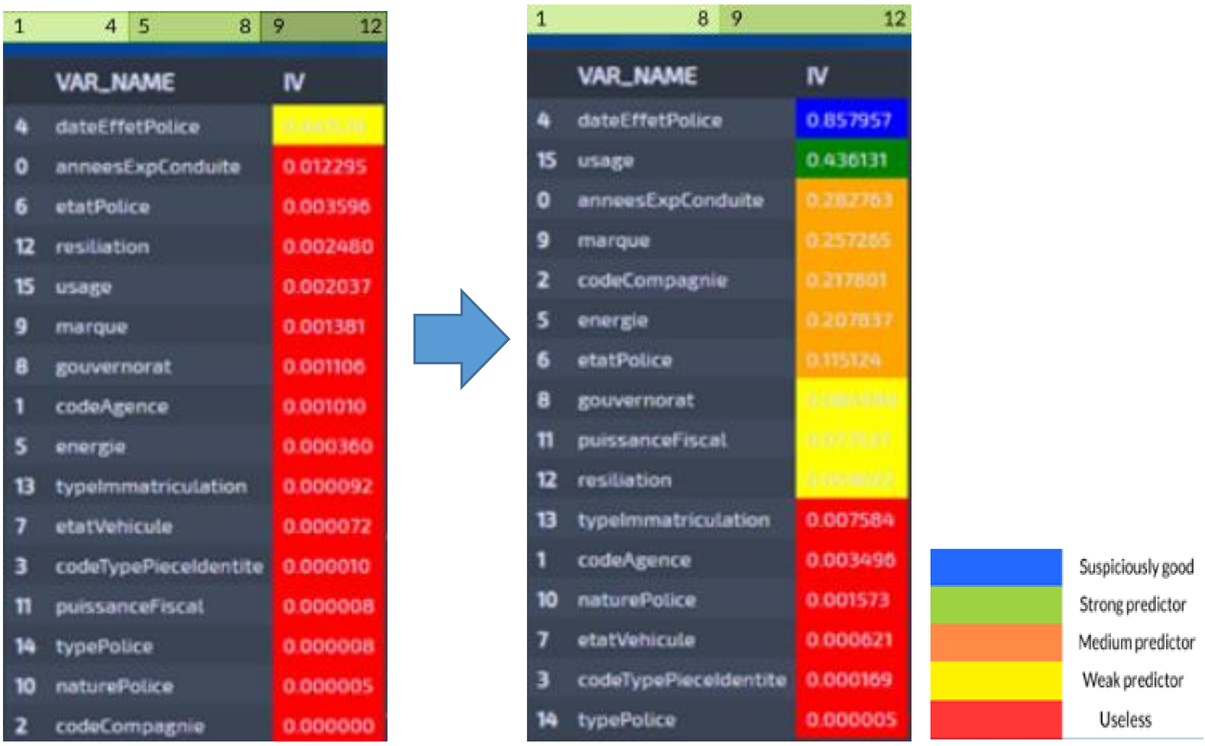


Figure 6: Sinistre relations

Before extracting the necessary features for our machine learning algorithms, we started off by cleaning our data by dropping null values and getting rid of columns inadequate to our models. We, also, generated new columns based on the information we had such as:
- The number of years of driving experience of each client
- Their home region
- A Boolean column that indicates whether their contract is terminated or not.

Our main goal is to classify clients. First, we considered classeBonusMalus as our target variable. However, the result was not satisfactory. Therefore, we created target variable that splits clients into 3 groups (Good, average and bad). After calculating the Information Value which is the measurement of the relationship power between a predictor and a dependent variable, we had to abandon this method and create a different target that splits clients into 2 groups (Good and bad) which led to our final table.



	VAR_NAME	IV
4	dateEffetPolice	[illegible]
0	anneesExpConduite	0.012295
6	etatPolice	0.003596
12	resiliation	0.002480
15	usage	0.002037
9	marque	0.001381
8	gouvernorat	0.001106
1	codeAgence	0.001010
5	energie	0.000360
13	typeImmatriculation	0.000092
7	etatVehicule	0.000072
3	codeTypePieceIdentite	0.000010
11	puissanceFiscal	0.000008
14	typePolice	0.000008
10	naturePolice	0.000005
2	codeCompagnie	0.000000

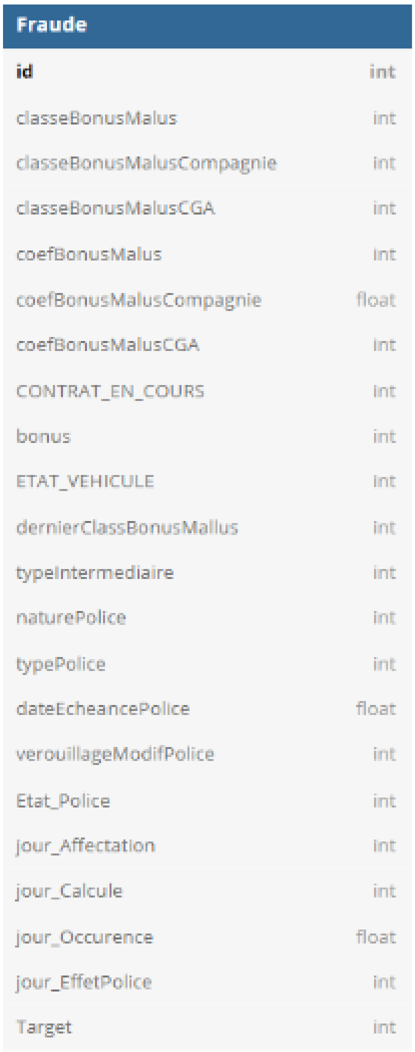
	VAR_NAME	IV
4	dateEffetPolice	0.857957
15	usage	0.436131
0	anneesExpConduite	0.282763
9	marque	0.257265
2	codeCompagnie	0.217801
5	energie	0.207837
6	etatPolice	0.115124
8	gouvernorat	[illegible]
11	puissanceFiscal	[illegible]
12	resiliation	[illegible]
13	typeImmatriculation	0.007584
1	codeAgence	0.003496
10	naturePolice	0.001573
7	etatVehicule	0.000621
3	codeTypePieceIdentite	0.000169
14	typePolice	0.000005

🟦	Suspiciously good
🟩	Strong predictor
🟧	Medium predictor
🟨	Weak predictor
🟥	Useless

Figure 7: IV (information value) tables with their color legend

2.2.2 Fraud detection:

For the fraud detection objective we have merged the two tables BonusMalus and Police in order to extract our 3 types of fraud which will be explained in the second part of the third chapter. After the data preparation we only left the following columns that we judged useful in our case.

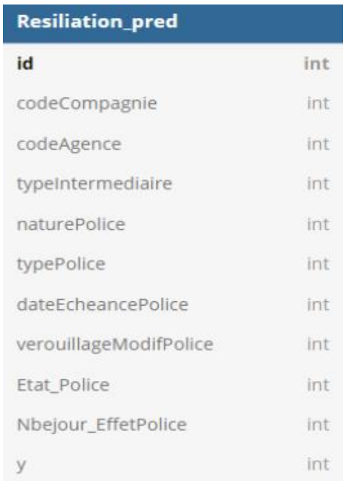


Fraude	
id	**int**
classeBonusMalus	int
classeBonusMalusCompagnie	int
classeBonusMalusCGA	int
coefBonusMalus	int
coefBonusMalusCompagnie	float
coefBonusMalusCGA	int
CONTRAT_EN_COURS	int
bonus	int
ETAT_VEHICULE	int
dernierClassBonusMallus	int
typeIntermediaire	int
naturePolice	int
typePolice	int
dateEcheancePolice	float
verouillageModifPolice	int
Etat_Police	int
jour_Affectation	int
jour_Calcule	int
jour_Occurence	float
jour_EffetPolice	int
Target	int

Figure 8: Fraud detection table

2.3.3 Contract termination:

In this section we only used the table Police after undergoing some data preparation and after the addition of a target variable:



Resiliation_pred	
id	**int**
codeCompagnie	int
codeAgence	int
typeIntermediaire	int
naturePolice	int
typePolice	int
dateEcheancePolice	int
verouillageModifPolice	int
Etat_Police	int
Nbejour_EffetPolice	int
y	int

Figure 8: Contract termination table

We chose not to add the BonusMalus table because there are a lot of clients that have many contracts in progress and the termination of one of them does not mean the termination of any others. As for the creation of the target, it was based on the column dateResiliation

2.3 Data visualization:

In this part we decided to dig more into details of the features we selected previously.

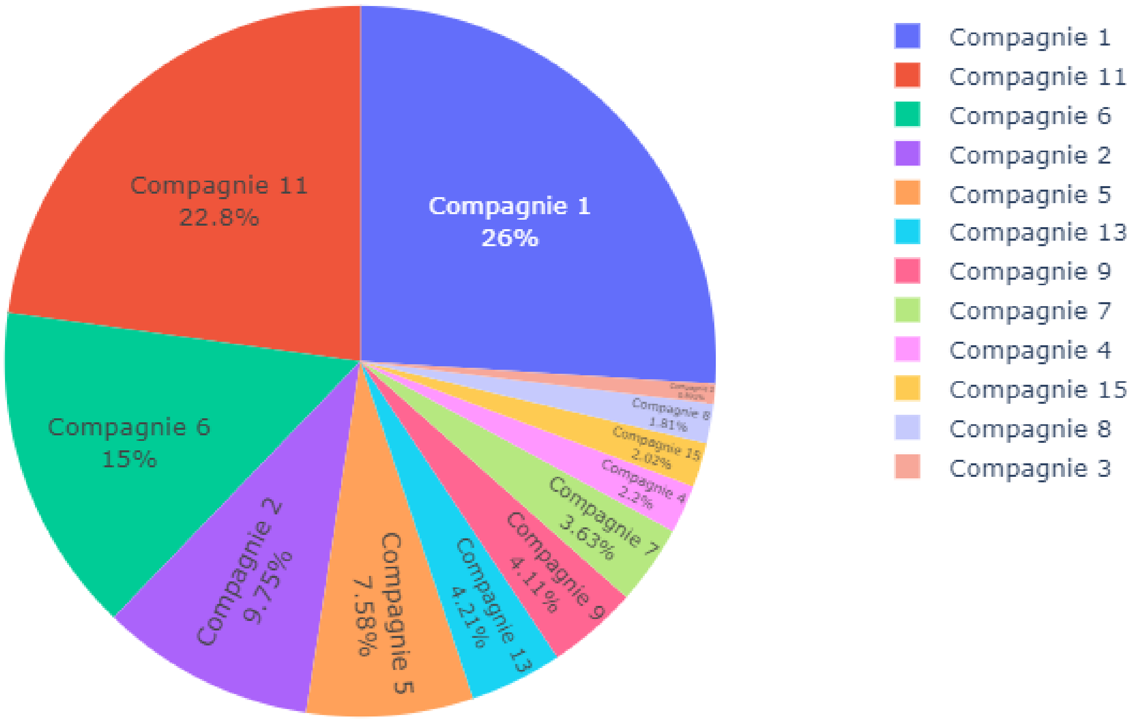


Figure 9: Police contracts distribution by company

For the feature "CodeCompagnie", the contracts shown below were mostly from the first Company with 26% of them.

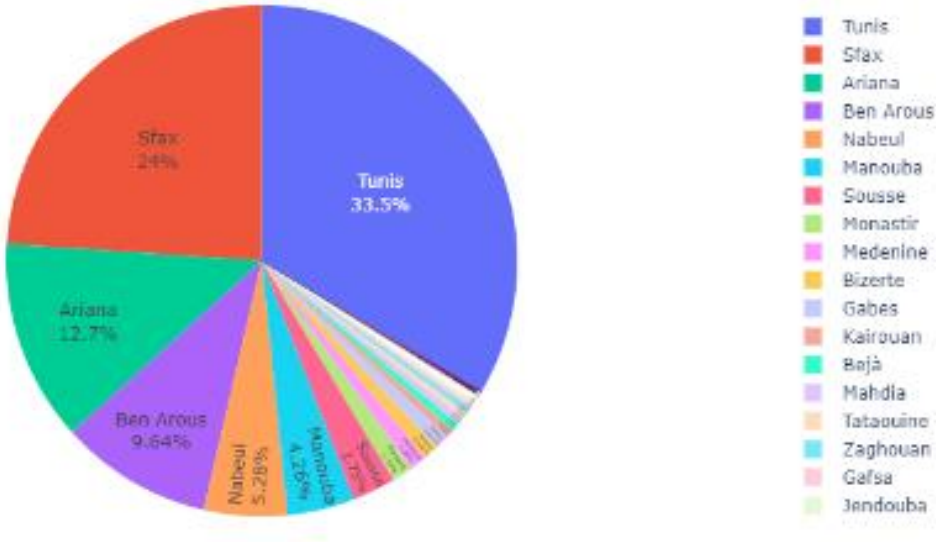


Figure 10: Police contracts repartition by company

For the 'gouvernorat' feature, Tunis was the most featured city in number of accidents happening.

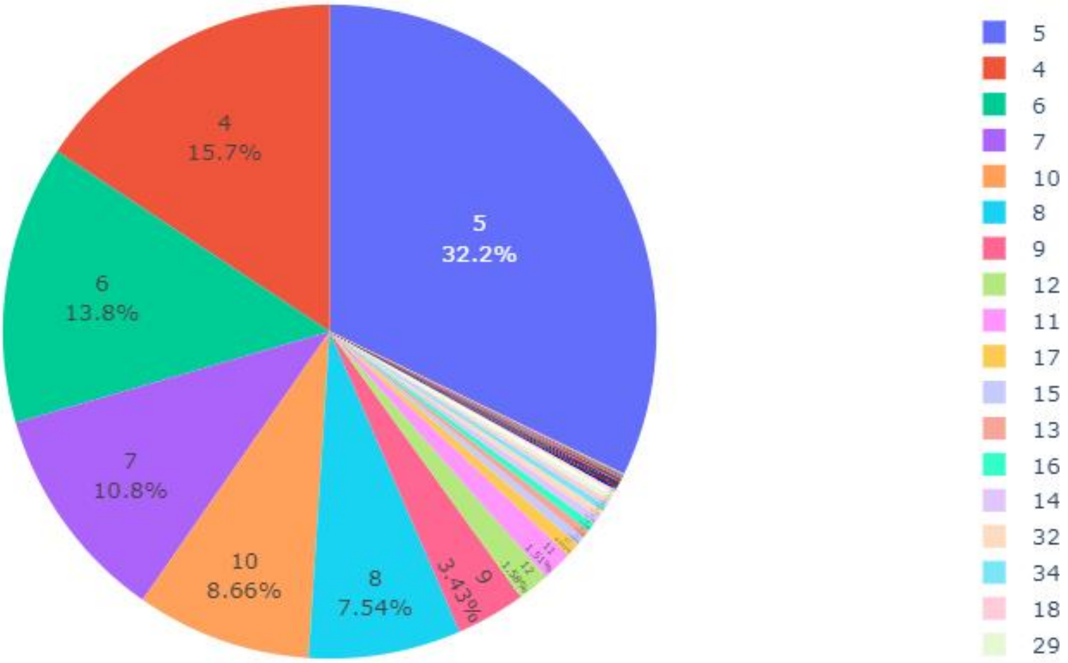


Figure 11: Number of accidents by vehicle horsepower

For the "PuissanceFiscal", 32.2% of the accidents happening involved a car with a horsepower of 5.

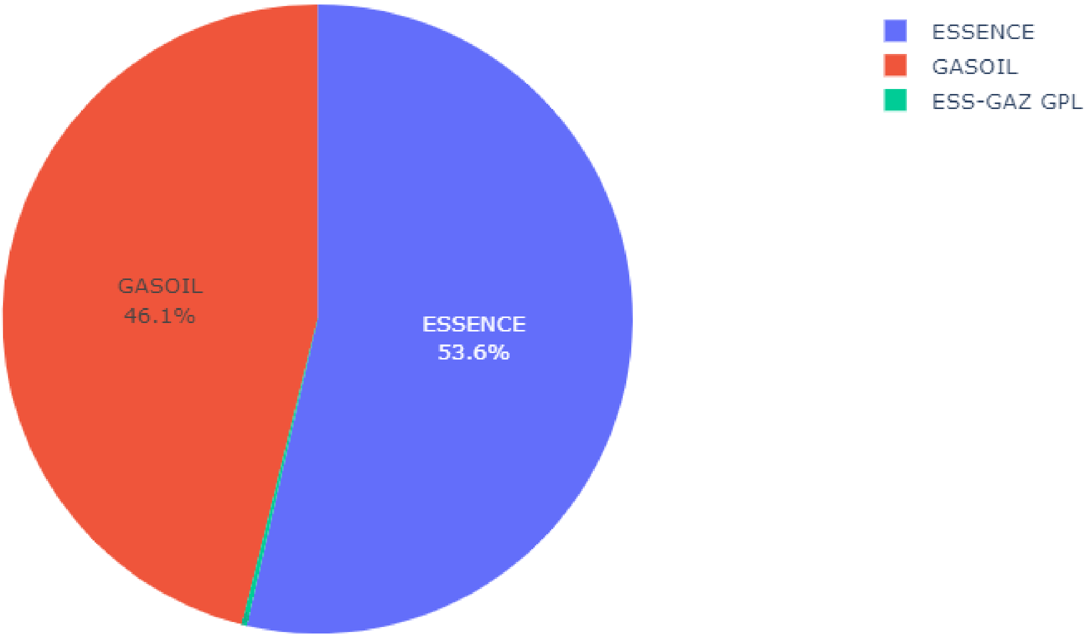


Figure 12: Number of accidents by vehicle energy type

The "Energie" feature presented a 50-50 situation between "ESSENCE" and "GAZOIL".

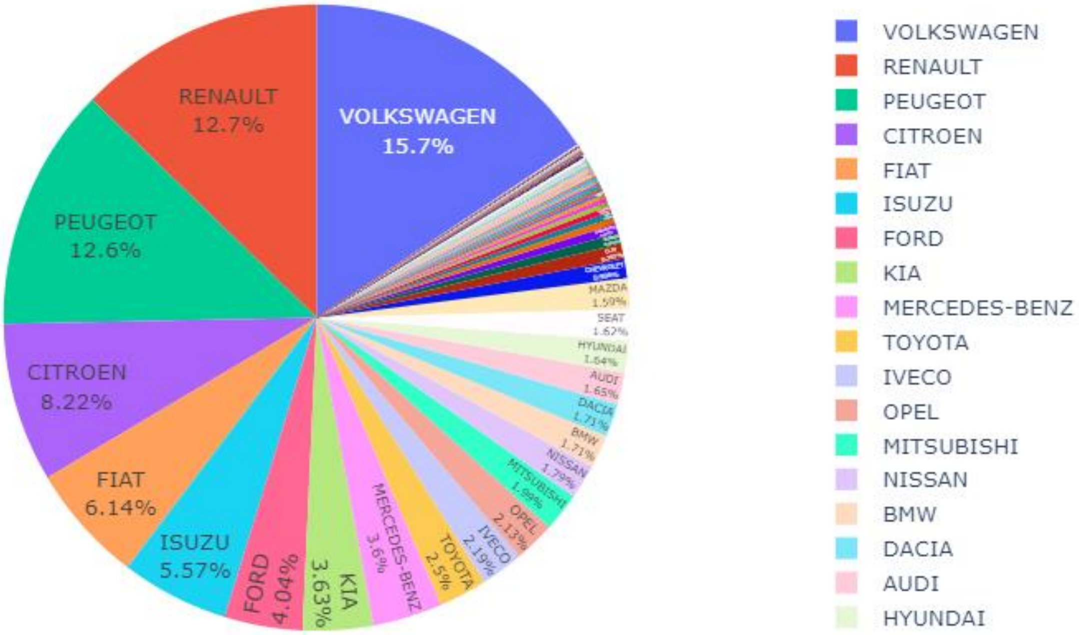


Figure 13: Number of accidents by vehicle brands

Furthermore, the majority of the accidents involved a Volkswagen, a Renault, a Peugeot, a Citroen and a Fiat.

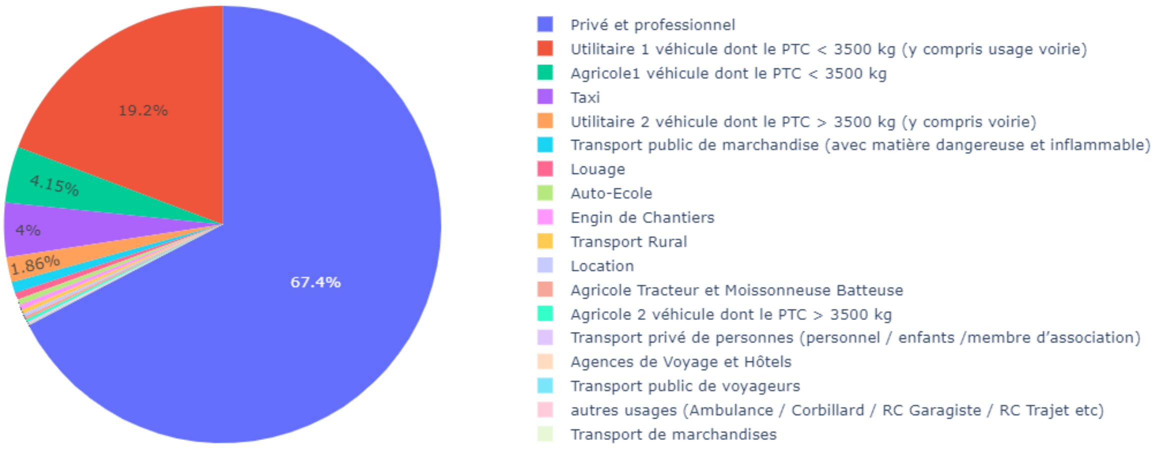


Figure 14: Vehicle usage distribution

The usage of those cars mentioned previously is mostly personal or professional (67.4%).

But all this data could be unfortunately misleading because when we look at both the number of clients from their experience in driving and the accidents those years of experience are involved in, we see that both graphs have the "31-40" part as a majority and could be meaning that our visualization is biased by the sampling of the data available for us and that pushed us to do some changes on our data before modeling.

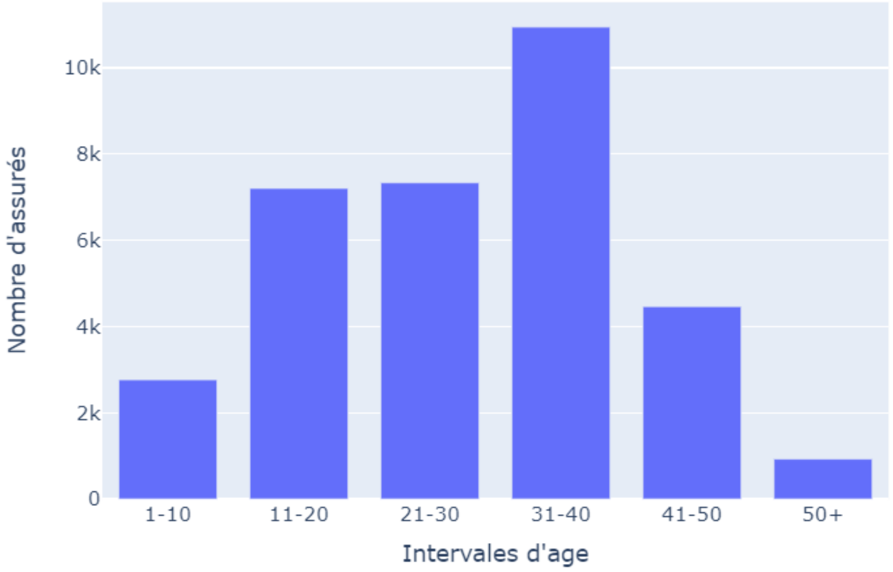


Figure 15: Number of clients by driving experience intervals

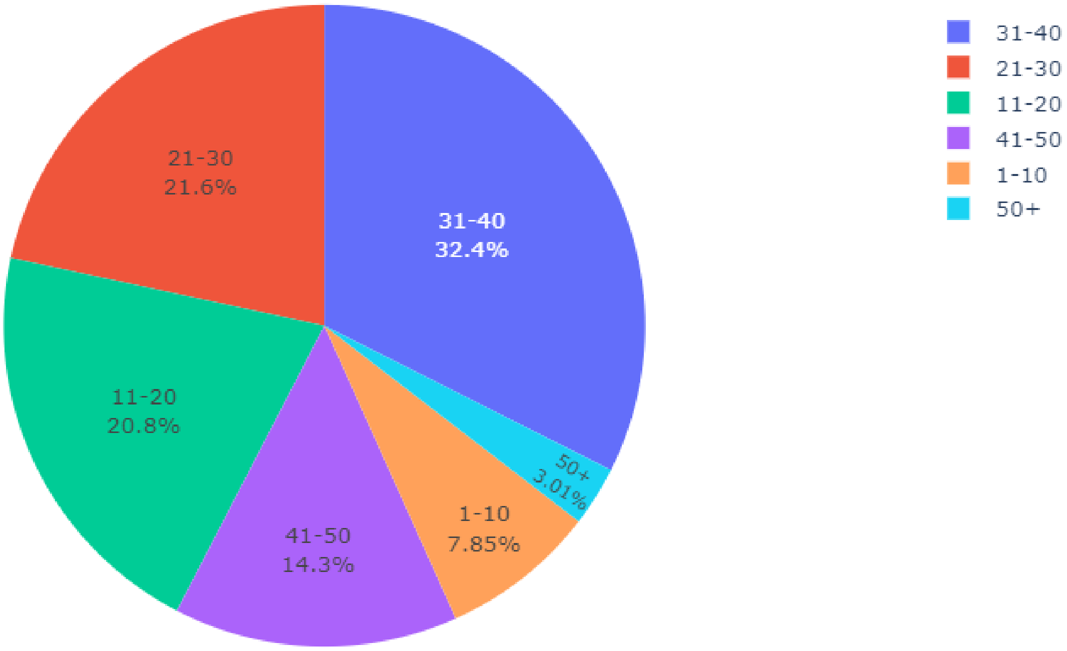


Figure 15: Number of accidents by driving experience intervals

CHAPTER III:

Data Modeling

In this chapter, we move to the core of the project which is the modeling of our data that will allow us to reach our primary goals: the client classification, the fraud detection and the contract termination.

3.1 Client Classification:

Before training our machine learning models, it looked necessary to try and balance the distribution of our target variable. The disproportion between both classes was significant: Class 1 represents almost 90% of our dataset. Hence, balancing our training data was necessary.

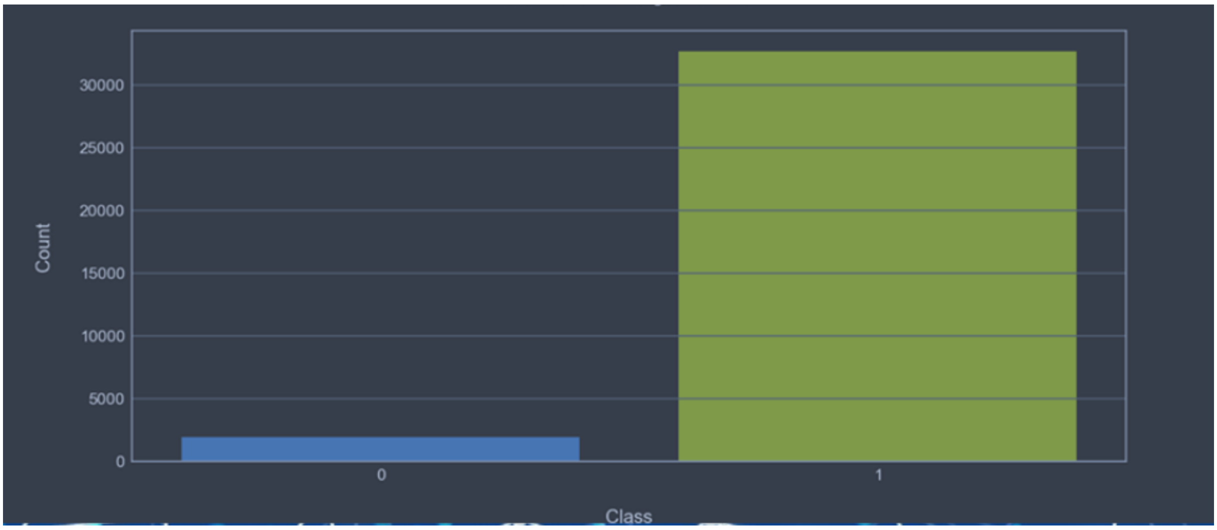


Figure 16: Distribution of target variable

Oversampling the minority with SMOTE was an option but given the huge imbalance, too much synthetic data can cause the model to diverge from actual information.

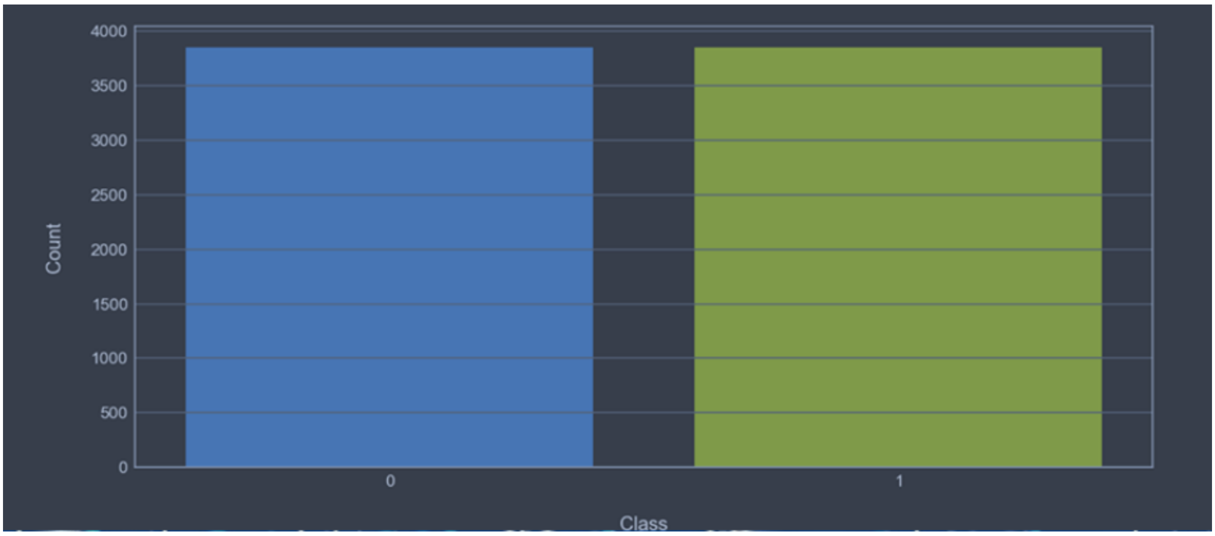


Figure 17: Distribution of target variable after OverSampling

Therefore, in order to minimize the creation of artificial data, down-sampling Class 1 to twice the volume of Class 0 was essential before using SMOTE to over-sample Class 0 and therefore end up with a balanced dataset.

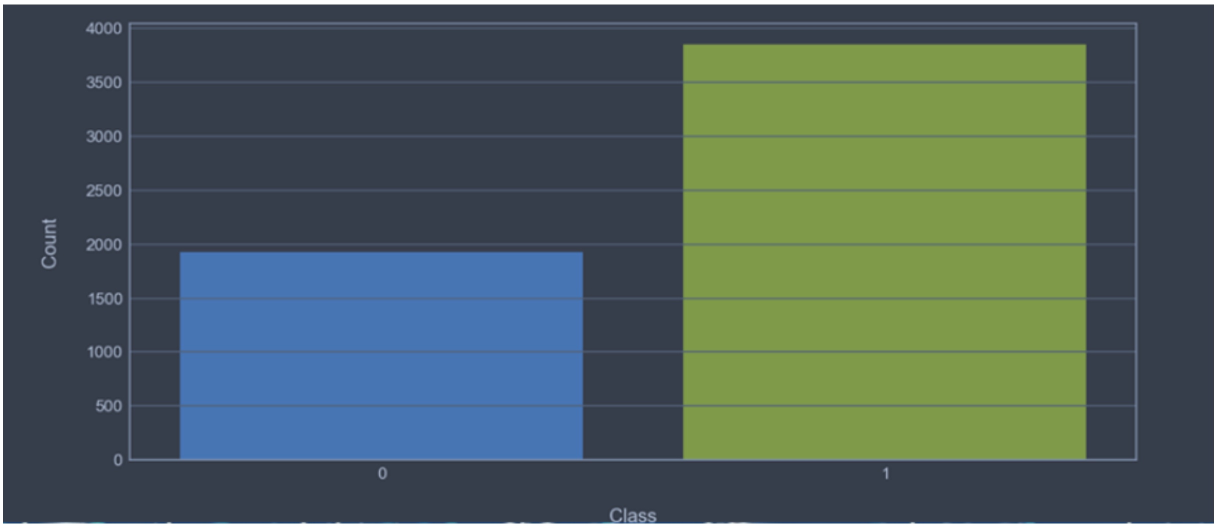


Figure 18: Distribution of target variable after DownSampling

We ran our data over five different machine learning models:

K-NN: K Nearest Neighbor is a very simple, easy to understand, versatile and one of the topmost machine learning algorithms. It is used for both classification and regression problems and is based on feature similarity approach.

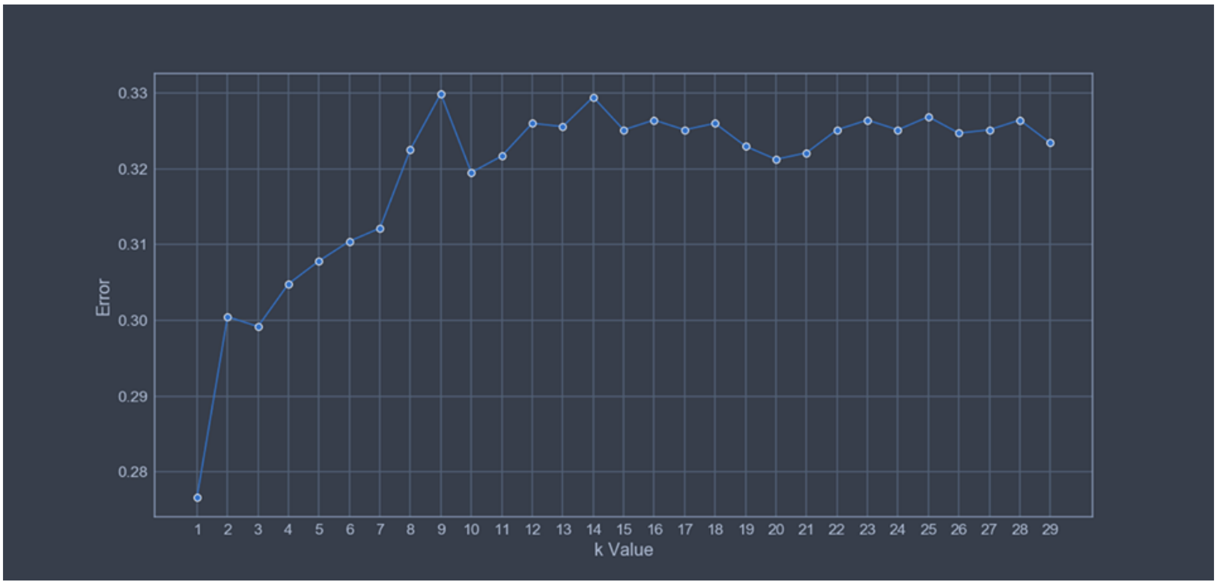
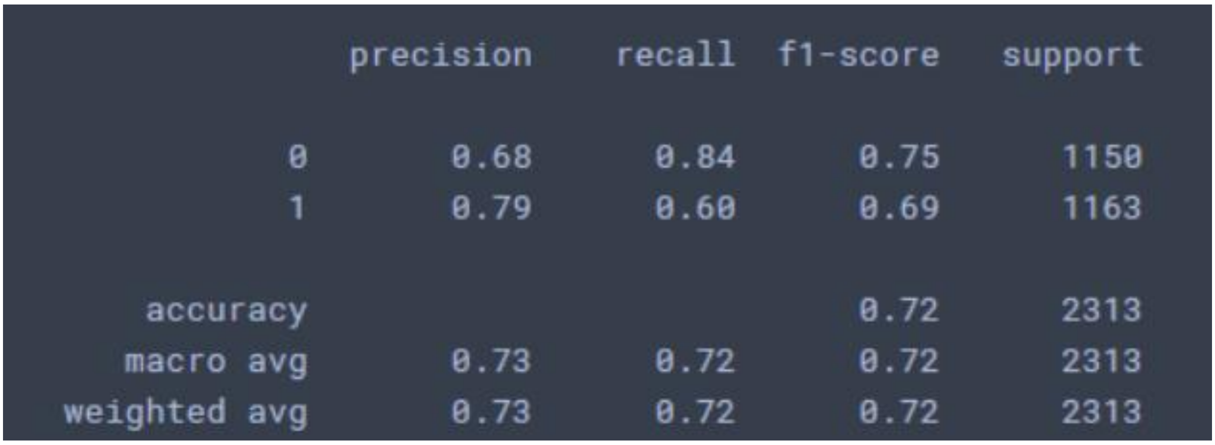


Figure 19: Error by K value on the KNN classification

Despite a relatively good precision and recall, our K-NN model looked unstable especially with an optimal number of neighbors equal to 1.

Logistic Regression: It is a supervised learning classification algorithm used to predict the probability of a target variable. The nature of target or dependent variable is dichotomous, which

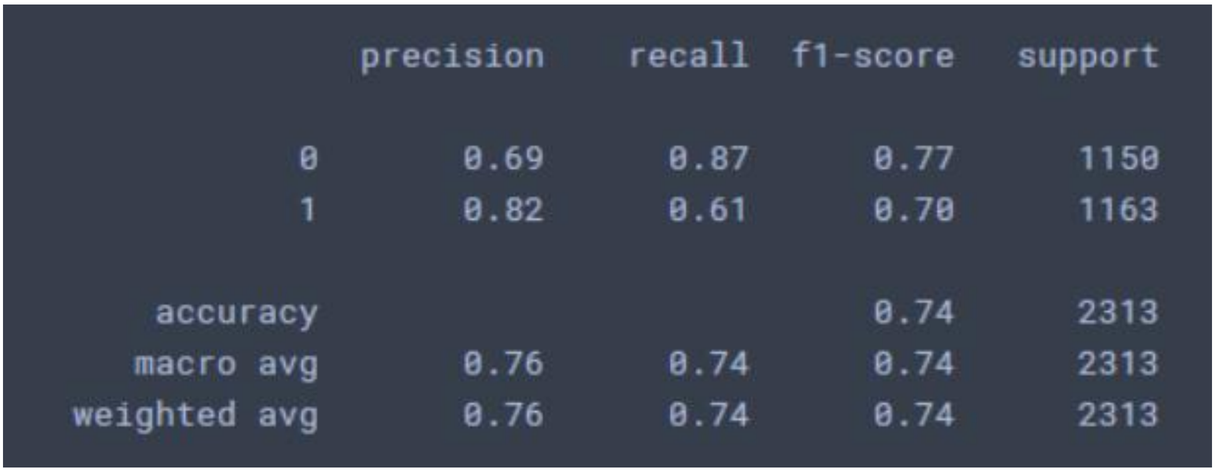
means there would be only two possible classes.
Mathematically, a logistic regression model predicts P(Y=1) as a function of X.



```
              precision    recall  f1-score   support

           0       0.68      0.84      0.75      1150
           1       0.79      0.60      0.69      1163

    accuracy                           0.72      2313
   macro avg       0.73      0.72      0.72      2313
weighted avg       0.73      0.72      0.72      2313
```

Figure 20: Logistic Regression classification report

SVM: It is a supervised machine learning algorithm which can be used for both classification and regression challenges. In the SVM algorithm, we plot each data item as a point in n-dimensional space (where n is number of features you have) with the value of each feature being the value of a particular coordinate. Then, we perform classification by finding the hyper-plane that differentiates the two classes very well.

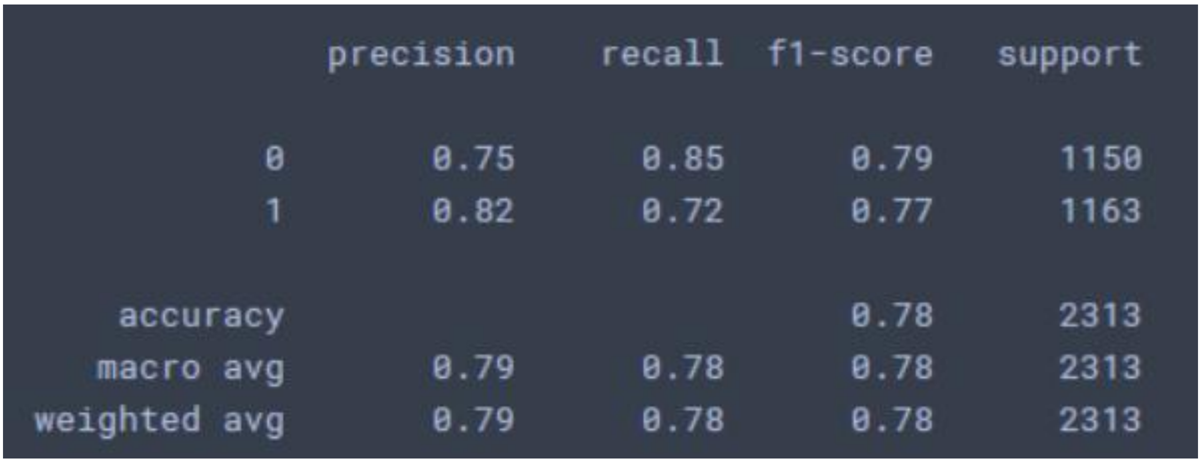


```
              precision    recall  f1-score   support

           0       0.69      0.87      0.77      1150
           1       0.82      0.61      0.70      1163

    accuracy                           0.74      2313
   macro avg       0.76      0.74      0.74      2313
weighted avg       0.76      0.74      0.74      2313
```

Figure 21: Logistic Regression classification report

Random Forest: Random forest, like its name implies, consists of a large number of individual decision trees that operate as an ensemble. Each individual tree in the random forest spits out a class prediction and the class with the most votes becomes our model’s prediction. The fundamental concept behind random forest is a simple but powerful one — the wisdom of crowds.

Naïve Bayes: It is a classification technique based on Bayes' theorem with an assumption of independence between predictors. In simple terms, a Naive Bayes classifier assumes that the presence of a particular feature in a class is unrelated to the presence of any other feature.



```
              precision    recall  f1-score   support

           0       0.75      0.85      0.79      1150
           1       0.82      0.72      0.77      1163

    accuracy                           0.78      2313
   macro avg       0.79      0.78      0.78      2313
weighted avg       0.79      0.78      0.78      2313
```

Figure 22: Random Forest classification report

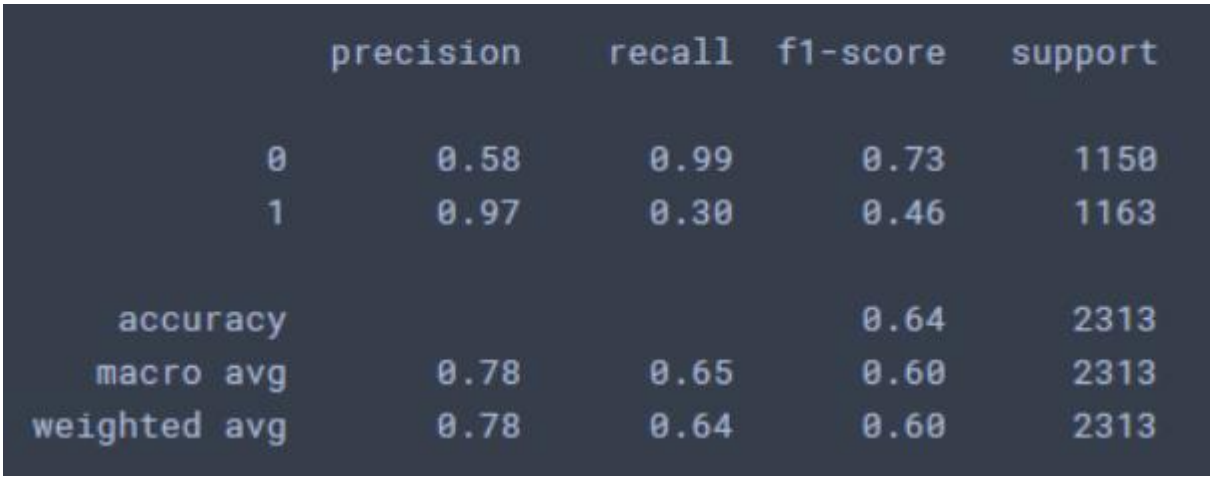
Bayes' theorem:

It states, for two events A and B, if we know the conditional probability of B given A and the probability of B, then it's possible to calculate the probability of B given A.

$$P(A|B) = \frac{P(B|A)P(A)}{P(B)}$$

Where A and B are events and .P(B) ≠ 0

- P(A | B) is a conditional probability: the likelihood of event A occurring given that B is true.
- P(B | A) is also a conditional probability: the likelihood of event B occurring given that A is true.
- P(A) and P(B) are the probabilities of observing A and B respectively; they are known as the marginal probability.



```
              precision    recall  f1-score   support

           0       0.58      0.99      0.73      1150
           1       0.97      0.30      0.46      1163

    accuracy                           0.64      2313
   macro avg       0.78      0.65      0.60      2313
weighted avg       0.78      0.64      0.60      2313
```

Figure 23: Naïve Bayes classification report

3.2 Fraud detection:

As was seen in the second chapter of this report specifically in the first section (data comprehension) there are three types of potential fraud in our data:

1. There are a lot vehicles (exactly 815) that have too many contracts in progress.As we can see here in this bar plot there are a lot of cars that they have 2 and even three contracts in same time

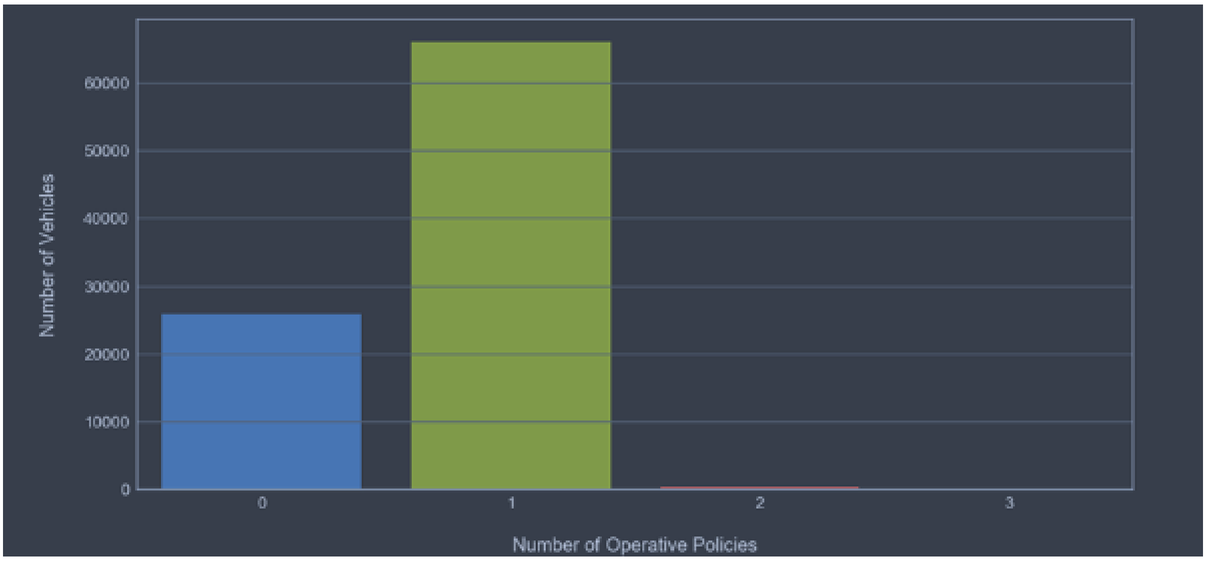


Figure 24: Number of vehicles by Number of operative contracts

2. The same vehicles are registered by many clients at the same time with an active contract
3. There are vehicles registered in multiple insurance companies at the same time

After running several models we chose the AdaBoost algorithm since it handed the best score:

AdaBoost: The basic concept behind Adaboost is to set the weights of classifiers and training the data sample in each iteration such that it ensures the accurate predictions of unusual observations. Any machine learning algorithm can be used as base classifier if it accepts weights on the training set

➔ With all this we created our model and after resampling data we get this graph with the classification report of our model.

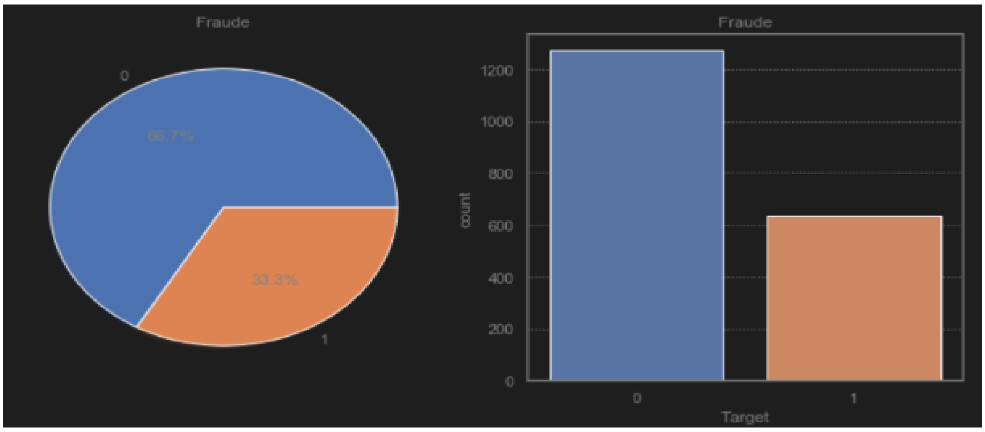
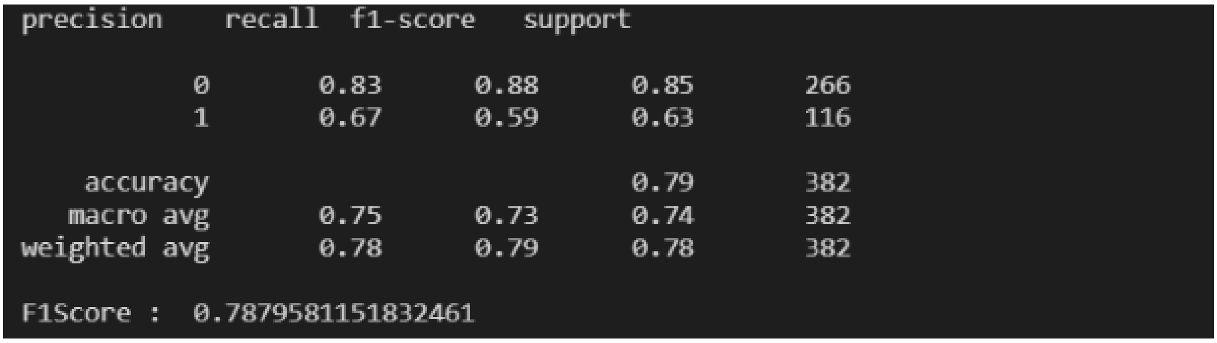


Figure 25: Fraud Count by Target variable

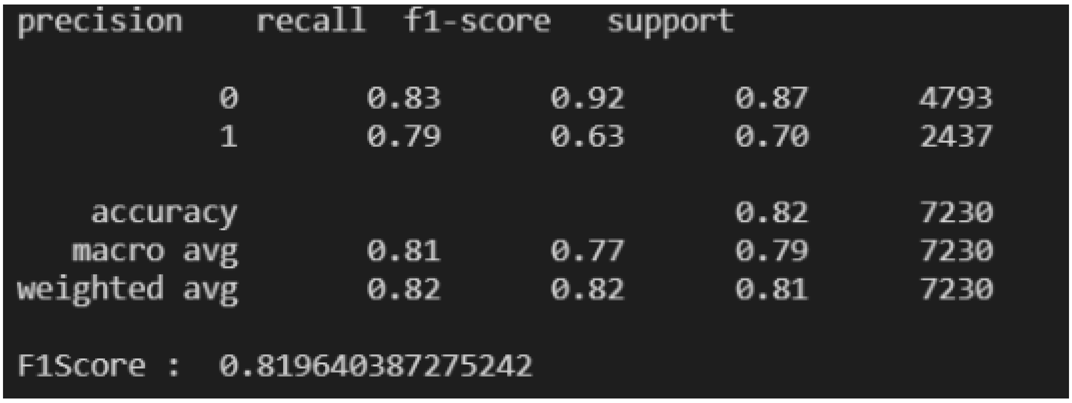


```
precision    recall  f1-score   support

           0       0.83      0.88      0.85       266
           1       0.67      0.59      0.63       116

    accuracy                           0.79       382
   macro avg       0.75      0.73      0.74       382
weighted avg       0.78      0.79      0.78       382

F1Score :  0.7879581151832461
```

Figure 26: AdaBoost classification report for fraud detection

3.3 Contract termination

Exactly like in the fraud detection modeling part, we applied the AdaBoost algorithm because it gave by far the best score



```
precision    recall  f1-score   support

           0       0.83      0.92      0.87      4793
           1       0.79      0.63      0.70      2437

    accuracy                           0.82      7230
   macro avg       0.81      0.77      0.79      7230
weighted avg       0.82      0.82      0.81      7230

F1Score :  0.819640387275242
```

Figure 27: AdaBoost classification report for contract termination detection

Conclusion:

Fraud detection goes beyond machine learning as graph theory is often used to identify clusters of insurance clients who are potentially simulating accidents in order to obtain their insurance premium. Such algorithm did not yield interesting results due to insufficient data. However, given the importance of such fraud, we decided to run the said algorithm on a generated dataset. We created a Graph using Networkx Python library where nodes represent insurance contracts and are connected with weighted edges if two were involved in the same sinister at least once. And as seen in the Graph on the right, the algorithm isolates cycles which could lead to detecting clusters of clients involved in insurance fraud.

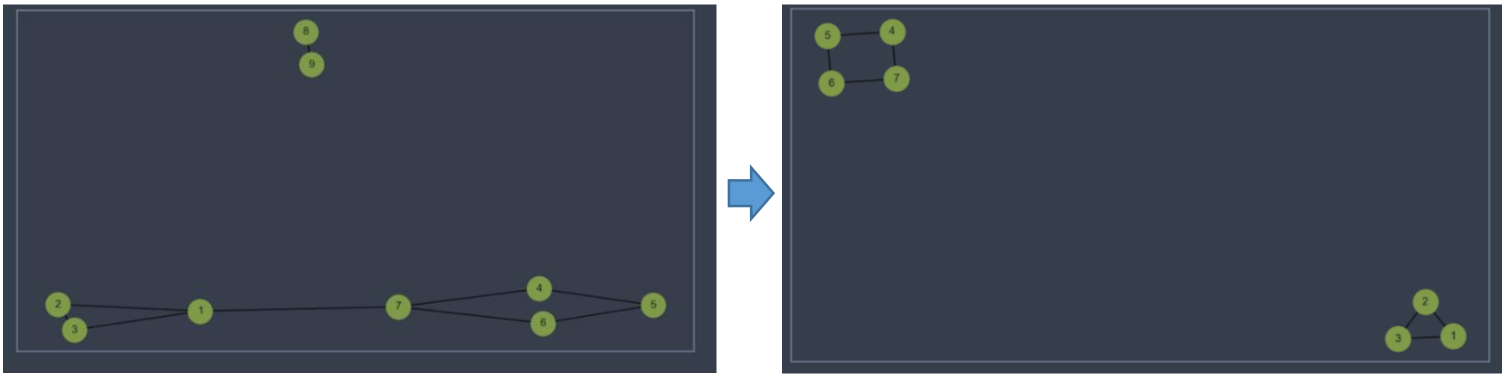


Figure 28: Fraud detection using graph theory

Most insurance companies process only 10–15 percent of the data they have access to most of which is structured data they house in traditional databases. That means they are not only failing to unlock value from their structured data, but also overlooking the valuable insights hidden in their unstructured data. Machine learning is one of the major keys to unlock important aspects of data and numerous insurance companies are starting to rely on AI to increase their profit such as Liberty Mutual, Progressive and Allstate