Data management project: report

Warning :
This project uses libraries/packages which are not pre-installed in Anaconda. Two solutions are provided, so that you can make work the code in the different notebooks:

1. You can simply install manually the two only packages we used which are not in the pre-installed Anadonda package, in your base environment or any virtual anaconda environment you want:

    a. Folium

    ```
    pip install folium
    ```

    b. Missingno

    ```
    pip install missingno
    ```

2. We created a *environment.yml* file, which contains all the packages we used in this project (does natively included in anaconda et does which are not). You can then create an anaconda virtual environment, which will have all the packages installed and be able run the jupyter notebooks, by running the following command in Anaconda prompt:

    ```
    conda env create --name your_env_name --file=environment.yml
    ```

Part 1: Data Loading & Representation

The Data Loading & Representation part consists in building an adequate representation of the data that was handed to us in csv format. This stage in the data management pipeline follows the data collection step and prepares for data analysis. A few tasks have been handled in this part:

- **Load the raw data in pandas**
- **Merge the different datasets into a single comprehensible and coherent dataset**
- **Handle missing values**
- **Reencode variables (nominations, categorization, types)**
- **Drop useless variables**
- **Construct a data representation of the geographical and administrative structure of France**
- **save the cleaned and reencoded data in csv format for analysis use.**

1) Data Loading in pandas

The people in charge with data collection handed us non organized data in csv format. In the data loading stage, we used pandas tool to load the csv format data, and we proceed to some reindexing of variables to prepare the merging stage.

2) Merging of the different datasets

We wanted to construct a single comprehensible and coherent dataset which would enable us to visualize and cross analyse easily all the variables available. To facilitate the merging process, we set as index the **'key'** variable (enables to identify data relating to a single individual) and applied the *join* method for "additional information" type variables (club, contract). Concerning geographical data, we used the **'insee_code'** variable to identify correctly data relating to a single city. We then applied the *merge* method (using the **'insee_code'** column) in order to obtain a full dataset containing all the variables available.

3) Handling missing values

Two variables in our dataset were containing missing values:

- **'CLUB'**: describes to which sport club the individual belongs to
- **'Contract'**: gives the working contract of the individual

Missing values are often tricky to deal with. Having missing values in a dataset can occur for different reasons:

- Error in the data collection stage
- Unavailability of the variable value for particular observations during the collection stage
- Bad storage of the data

Given the high numbers of missing values for each variable, missing value handling had to be done with great care, since it could then have huge consequences on our later analysis.
They were several possible ways for us to handle those missing values:

- Drop the variable altogether
- Drop the observations which had a missing value
- Use domain knowledge and information provided by other variables to unveil the meaning of those missing values

- Concerning the **'Contract'** variable, using economic domain knowledge as well as using the information provided by other variables, we figured out that missing values in **'contract'** came from the fact that not all individuals in our sample were active workers. Many were either unemployed, retirees, housewives, or students etc. and hence had no working contract. We filled missing values by creating two new categories ***unemployed*** and ***no contract*** (economically inactive).
- Concerning the **'Club'** variable we figured out that the large number of missing values was probably resulting from the fact that many people were not part of any club.

4) Reencoding variables

The reencoding of the variables was performed through three different parts:

1. Provide new nominations to categories
2. Recategorize categorical variables
3. Assign some relevant data types to the variables

New nominations:
The data obtained from the collection phase was messy, full of abbreviations and unclear nominations. The first step of variable encoding consisted in providing some clear nominations to

each category of each categorical variable. This process facilitates interpretation of results in the data analysis phase.

Recategorization:

The dataset contains many categorical variables, some of which contain a lot of categories. The more category a variable contains the more precise the information it unveils. However, having too many categories can prevent us from making useful interpretations in the data analysis stage. (especially when categories only contain a very small number of observations)
Several techniques were used to recategorize.

Reencoding of the arrondissements, and their populations:

We observed that it was not possible to study well the size of the big metropoles (Paris, Lyon, and Marseille), because the INSEE_CODE was linked to the arrondissement, and not to the cities. Hence, we decided to reencode the arrondissement, and the INHABITANTS value of those arrondissement. In order to reencode, we used the method **.replace(),** on the variable 'Nom de la commune', to replace the name of the arrondissement by the name of the city. To replace the population of the arrondissement by the population of the city we filtered our dataset using the variable 'Nom de la commune' and the method **.startswith()**, to replace 'INHABITANTS' by the sum of the population of all the arrondissements of the city.

To reencode the variable « act »:
we used the theory of the INSIDERS/OUTSIDERS, from the new Keynesian Economic. These nominations concern only employees: hence we kept away a category « independent ». Additionally, we verified with confidence interval the relevance of our reencoding. (see part II, Predictive part, about robustness)

Data types:

Pandas provide useful built-in data types to improve the working experience when manipulating different kinds of data. In our dataset we work with many categorical variables, as well as certain boolean variables (created from recategorization). Using some code, we identified the different kinds of variables (numerical, boolean, categorical) and assign them the appropriate built-in data type.

5) Dropping variables

The Feature selection stage (stage where we select the most useful variables in order to predict the results of the marketing campaign) follows the data analysis stage. However here we can already identify some variables which might be helpless.
For instance we decided to the drop the **'OCCUPATION_24'** variable, since the detailed information it provided could already be accessed through simultaneously using the information in **'act'** (activity),'**Occupation_8'** and '**contract'**.

6) Data Representation of geographical & administrative structure of

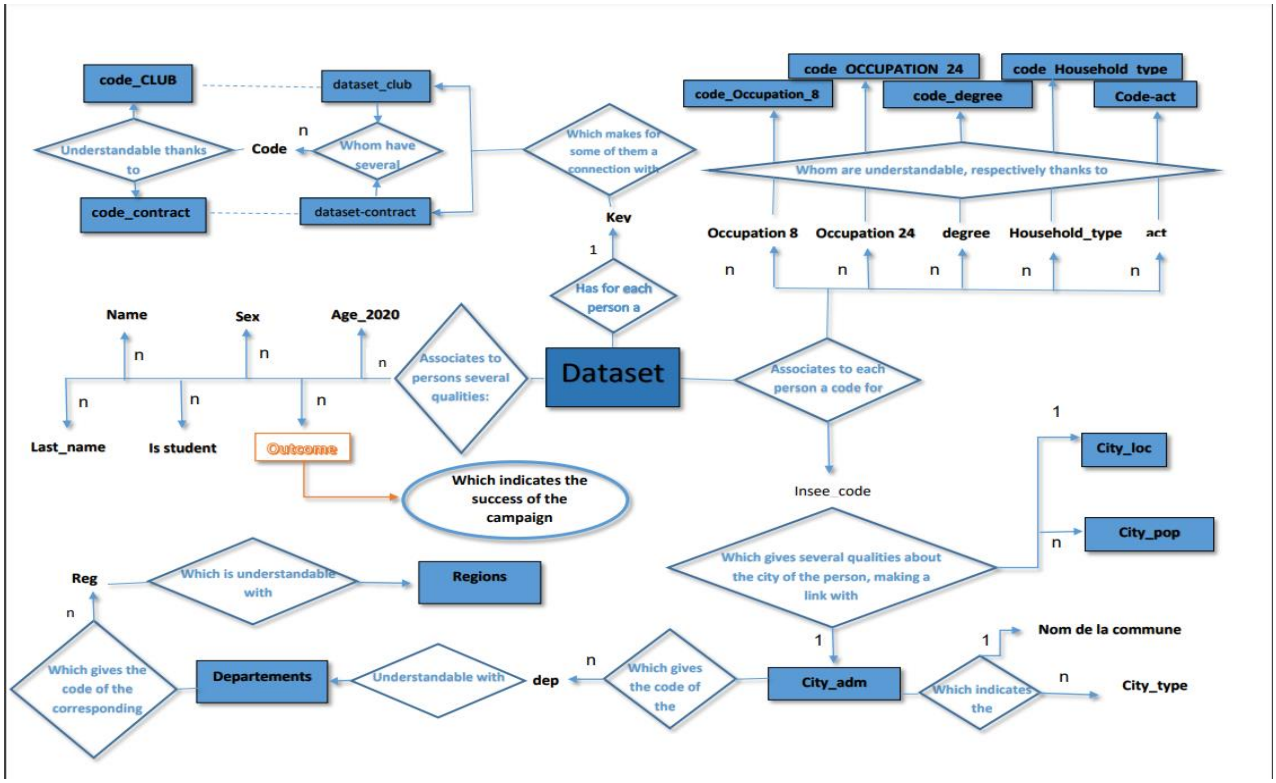
France

Using the *set_index* method of pandas, we constructed a dataframe by which one can simply access all the informations related to a city in France. The Data representation replicates the administrative structure of France.

7) Save cleaned data

We then saved the dataframe containing the information relevant for the analysis of the Marketing campaign, as well as the dataframe representing the geographical and administrative structure of france in csv format.
In depth explanations for each data manipulations undertook are provided along the coding cells. This report simply aims at providing a global understanding of the work accomplished in the loading and representation stage.

ER diagram



Part 2: Person level analysis

The descriptive and predictive part consists in finding which category of variables display particularly high success rates. The goal is to identify high potential prospects for the future marketing campaigns of the company.
Here are several tasks handled in this part:

1) A descriptive part

- find the success rate of the global sample

- create some new categories by reencoding, to make more readable some variables
- describe the typical profile of the success and failure sample

2) **A predictive part**
- Computation of the frequencies of the categories of the several variables, and comparison to the global success rate
- using a confidence interval to verify the robustness of our results
- construction of a machine learning algorithm, to aim at predicting the outcome of the Marketing campaign, and to precise the feature selection made in the descriptive part.

3) **Conclusion**

I. Descriptive approach:

The objective is here to identify characteristics of the persons in the “success” set that differentiate them from the “failure” set.

A. Success rate of the global sample:

We studied the general success and failure rate of the sample, by using the method **.value_count()**, apply to 'Outcome' in the data frame.
We found the following result: a success rate of 57,2% and a failure rate of 42,8%.

B. The final reencoding:

Create new categories to make readable the variables with number

We observed very little correlation between numerical variables, and the 'outcome” variable, with the method .corr().
However, that does not mean that we cannot observe trend depending on those variables. We can only from this analysis the absence of linear correlation (we do not observe linear relations such as: the younger people are the higher their success rate is).

This action concerned two variables:
- **age_2020**
- **INHABITANTS**

To study the age, we chose to use the method. cut**()**, apply to **age_2020**, to get the results by decades, from 15 to 95 (15 was the age minimum in the sample, and there were only few people over 95 years old), by creating a new variable **agecat**.

To study the variable INHABITANTS. We chose to create a new variable “**aire_urbaine**”. In order to reencode those category, we created a reencoding function “reencode_aire_urbaine” to apply it on the variable 'INHABITANTS', by using the method **.apply()** and the **lambda** function.
The new categories are:
- **'aire rurale'** : from 0 to 2000 habitants in the city
- **'petite ville'**: from less than 2,001 to 20 000

- **'ville moyenne'**: from 20,001 to 100,000
- **'grande ville'**: from 100,001 to 1,500,000
- '**Paris**' more than 1,500,000

Quick geographical science disgression:
Why did we choose those new categories, and how did we choose the following size of populations to do the reencoding?
According to main geographical works, and reports of INSEE, some dynamics can be observed, in a certain way correlated with the size of the city. Hence, we considered we could determine some categories:

-"**aire rurale**" are defined as commune which are not considered as city by the Insee (with less than 2000 habitants): this category is not the same as the usual category used by INSEE: "zone à dominance rurale", which is defined by the number of jobs in an area (INSEE uses the expression "bassin d'emploi"). Of course, we do not get those data, which are considered as most relevant. But we can suppose, due to geographical works, that those villages, are most of the time in "zone rurale".

-"**petite ville**": to speak about communes considered by the INSEE as cities, but which are normally not define as "ville moyenne. In main works, dynamics studied are the same that the dynamics of "zone rurales".

-"**ville moyenne**": defining this category is not easy. Indeed, this definition can vary a lot depending on the authors, in the geographical literature. that can begin from 20 000 habitants, 30 000 or even 40 000, and be limited by 100,000, or 200,000. The notion of "ville moyenne" depends more on jobs, economic or cultural fonctions (cf "LA NOTION DE "VILLE MOYENNE" EN FRANCE, EN ESPAGNE ET AU ROYAUME-UNI", FROM FREDERIC SANTAMARIA). It's of course not really relevant to class in the same category "Levallois-Perret", which benefit from the same dynamics of the West of Paris, than other "villes moyennes" in the "diagonale du vide". We are aware of those critics, but we assume doing this choice almost arbitrarly.

-"**grande villes**": we can begin to speak of "centre". Many works consider that positive dynamics (demographical, economic, cultural...) start often from those level of city's size.

-" **Paris**": we did a difference with other "grande ville", due to the strong economic and political centralisation of power in Paris (the famous "macrocephaly parisienne").

As we said, this variable is not precise enough to speak about "centre" and "periphery", and about "metropoles", even if these notions are in a kind of way linked to the size of the population. We know we get here an imprecise idea of those notions ... To correctly do this study (which could be probably interesting) we would need to get information about "reseaux" between cities, "relations de dépendance", "aires d'influence" or number of jobs in those areas.

C. Analysis of the success sample and of the failure sample

We studied here the categories which were the most represented in both samples, in order to get the typical profile, in both sets.
Indeed, for each variable, we studied the proportion of several categories in both sample (success and failure): To do a long story short, we used the method **pivot_table**, to differentiate people in the success and the failure set, and we divided the number of people in each category in both sets by the total number of each set (failure and success).

To observe the results, we plotted bar diagrams.

D. Summary

The typical profile in the success set is:

- a woman
- not student
- economically inactive or “employé”
- “actif”
- with a degree “inferieur au baccalaureat”
- insider (as we defined with our new categories), or without a contract
- without a club
- having a family
- leaving in a “commune simple” or in a “chef lieu canton”
- “young” (less than 55 years old)
- leaving in a petite ville, or in a ville moyenne

the typical profile in the failure set is:

- being a man
- not being a student
- being retired
- not having a club
- not having a contract
- being « actif avec un contrat ou en stage rémunéré », or « retraité ou en pré-retraite »
- having a degree “inferieur au baccalaureat”
- having a family
- being aged from 55 to 75 years old
- coming from a “commune simple”
- coming from a petite ville, or from a grande ville

We can do several observations, and critics, about the relevance of this approach:

- some categories seem in contradiction with other: for example in the success sample,“actif” (from the variable act) and “sans activité professionnelle” (from the variable OCCUPATION_8) are dominant categories in the success sample, depending on different variable: that's probably due to the fact that in the variable “act”, “actif” is composed of individuals which are divided in several categories in the variable CSP, and so have less importance face to the category “sans activité professionnelle” (which is composed of people from several categories in the variable “act”)
- not being a student is the typical profile of both sample: we can suppose that this observation stems from the fact that most people in the sample are not student.

We can then only apprehend the strict composition of those sample.

II. PREDICTIVE APPROACH

1) Analyse of the frequencies of the categories of the several variable

The objective is here to get the frequencies of every categories, of each variable, to know if being part of a particular category gives more chance to be part of the success or failure sample.

For each variable, we used the methods **pivot_table, numpy_array** and the function **round**.

We used bar graphics to get an clear view of the frequencies, by using the method **.plot()**.
The objective was here to compare the frequencies with the global success rate. A result is interesting, on the predictive point of view, if it is over the global success rate: 57,2%. Indeed, there is more success in the sample. We cannot then compare the results of the categories to 50% to have interesting results to trace people with a lot of chance of success.

To complete the analyse of frequencies of the numerical variables (age_2020 and INHABITANTS), we also chose to study the correlation, thanks to method **.corr()** with the variable 'Outcome':

- we observed a low correlation for age_cat
- we observed a decorrelation for INHABITANTS

That does not mean we cannot see interesting results for both of those variables: the decorrelation is just due to the fact that we don't have linear correlation.

We chose to eliminate of our study variable homogeneous results across categories: it was the case of **IS_CLUB** and **IS_STUDENT**. IS_STUDENT get more manifest results, but not so much, and is really linked to other categories from other variables.

This method is different than the method we adopted in the descriptive part: Indeed, results in the descriptive part can be due to the share of a specific category in the sample. That explain the difference of results.

Remarque: again, we didn't study the variable Region and departments in this part: observations would be the same as in the **grouped analysis part** but explained in a better way. (the north/south disparity, and the bad robustness for a lots of departments)

**We sum up our results in the table, in the part "summary of the results"

2) Cheeking the robustness of the results by using a confidence interval

Despite our reencoding in the part 1, we are not sure that all our results are perfectly robust, due to the size of our sample. Hence, we decided to study rigorously all of the results, for all the variables.

We built a function, to get a 95% confidence interval of all the frequencies
The principle is the following:

- using a loop to study all the categories
- We have to check if the "len" of the category is >30 or <30, to know if we use a **normal or a t distribution.**

If the confidence interval was too big, we renounced to study the category. We chose for example this option with the category "Doctorat de recherche", from the variable "degree".

3) Summary of the results

Irrelevant variables :

- IS_CLUB
- IS_STUDENT

+ : 15% over average

++ : 20% over average

+++ : 30% over average

- : 10% under average

-- : 20% under average

	More chance of failure than the average	Same average than the sample	More chance of success than the average
City_type	Prefecture de région	Commune simple	Paris Chef lieu canton Sous prefecture
Csp : « Occupation_8 »	« Ouvriers »(--) « Retraités »(--)	« Employés »	« Agriculteurs-exploitants » (+++) « Profession intermédiaires » (++) « Artisans, commerçants et chefs d'entreprise »(++) « Cadres et professions intellectuelles supérieures »(++) « Autres personnes sans activité professionnelle»(+)
Gender : « Sex »	« Male »		« Female »
Activity : « act »	« Retraités ou pré-retraités »(--)	« actifs ayant un emploi, y compris sous apprentissage ou en stage rémunéré » « Chômeurs »	« Femmes ou hommes au foyer » (++) « Autres inactifs » (++) « Elèves, étudiants, stagiaires non rémunéré de 14 ans ou plus »
« Contract »	« no contract » (-)	« outsiders »	independants(++) insiders
« Household_type »	« célibataire » (--)	« famille »	colocataire (+) famille monoparentale (+)
Agecat («age_2020 »)	Plus de 55 ans (pourcentage d'echec de la campagne augmente avec l'âge)	Moins de 25 ans Entre 45 et 55 ans	Entre 25 et 45 ans
Size of the city : « aire_urbaine » (« INHABITANT »)	« grande ville » (-)	« aire rurale » « petite ville »	« Paris » « ville moyenne »
Degree	« diplôme inférieur au baccalauréat » (-) « Baccalauréat général ou technologique,	« BAC+3 ou BAC+4 » « Doctorat de recherche »	« sans diplôme » « BAC+2 » (+) sans diplôme « BAC+5 » (++)

	diplôme équivalent » (-)	(results are not robust)	« Baccalaureat ou brevet professionnel, diplôme équivalent » (+)

4) Construction of a machine learning algorithm

In this subpart of Person level analysis, we construct a machine learning algorithm. This modeling serves two purposes:

1. First, we aim at predicting the outcome of the Marketing campaign for prospects using the insights we draw from the descriptive and predictive analysis made above.
2. Second, we intend to precise the feature selection made in the descriptive part. In fact, in the descriptive part, we ended up identifying the variables that had a significant impact on the marketing campaign outcome. However, we did not yet explore the possible correlations existing among the numerous categorical variables. Running a machine learning algorithm using all the variables selected from our descriptive analysis, allows us to then measure the predictive power of each of those features. We then end up with a restrained subset of relevant variables for predictive usage.

In this prediction and modeling analysis, we proceed in a few steps:

1. We start by building a simple benchmark model to use its predictive power as a reference point.
2. Then we build a more ambitious model using all the variables we identified as possibly useful from the descriptive analysis.
3. Finally, we run a feature selection algorithm, which identifies the variables with the highest predictive power. We then restrain the previous model only using the variables selected. We test the prediction accuracy of the final model and compare it to the benchmark's accuracy score.

Benchmark model:

As a benchmark model, we constructed a basic logistic regression using a reduced set of variables: age, sex, north/south localization, and contract. We restrained ourselves to those variables because of their significance (identified in the above descriptive analysis), and because we expected those variables would be poorly correlated. This benchmark model scored a 70.7% accuracy score (on the test set), which means that it correctly predicted the marketing campaign outcome for 7 out of 10 new prospects. This result constitutes a good starting point.

NB: ''is_south" variable

We included in our prediction analysis and modeling the 'is_south' variable, describing whether the prospect is located in the north or south of France. This variable was identified as particularly significant when performing the grouped analysis in section 2.3.

Model with full set of variables:

We then created a set of models using all the variables we identified as significantly impacting the outcome of the marketing campaign in our descriptive analysis. First, we built a logistic regression, which performed terribly (accuracy score: 56%) tending to confirm the presence of highly correlated variables among our categorical variables (Logistic regression is highly impacted by correlation among explanatory variables). Then we used a Decision Tree (without any parameter tuning). This choice is justified by the fact that our set of features contains a disproportionate amount of categorical and Boolean variables. Decision Tree performs well on categorical variables and is immune to multicollinearity which makes it especially well-suited for our case. The model

performed an accuracy score of 75.1 % on the test set.

Feature selection and final Model:

Using the Decision Tree model with full set of variables, we evaluated the relative predictability importance of our features. Based on a bar plot displaying the relative feature importance's we used a threshold to select a subset of the most relevant variables. This feature selection enabled us to drop from our modeling correlated variables that were not adding additional information in the predictive stage. We ended up constructing a final Decision Tree model only using the following variables: *age, number of inhabitants, sex, north/south localization, CSP (Occupation_8), degree, type of the city*.

Hence, these are the variables we consider the better suited to predict the outcomes of future marketing campaigns. This model scored 77.68 % of accuracy on the test set (after correcting for overfitting, see NB), showing a very slight increase in performance while using way fewer variables (compared to the Decision Tree with full set of variables).

NB: Decision Tree and overfitting

A Decision Tree is an algorithm that uses the different variables/attributes available in the dataset in order to infer a set of rules which will help classify each observation to the different categories of the dependent variable. More precisely this algorithm uses attributes to split the data into subsets until each subset is pure (i.e., contains only observations belonging to a single category of the dependent variable). This mechanism results in building a decision tree (set of nodes that ask a binary question i.e., is the individual a male or a female?) through which each new observation gets assigned to a specific category based on its attributes (prediction);

However, because decision trees are by default designed to split the dataset until pure subsets are formed, they naturally lead to overfitting. By adding layers of deepness and splitting over and over until perfectly separating observations of each dependent variable categories, the algorithm might catch some noise specific to the training dataset and fail to unveil the interesting underlying causal links. We call this phenomenon overfitting since the algorithm fits the data of the training set too well. It is often characterized by poor generalization, i.e., poor performance on the test set.

In our modeling, using a decision tree, we experienced some very high accuracy scores on the training set (99%) linked to much poorer results on the test set (around 70%). This indicated overfitting. We hence used a post-pruning technic, by optimizing the parameterization of the cost complexity parameter. Post-pruning means to run a decision tree, and to then try to correct its overfitting by reducing its deepness (cutting its lower branches). We hence iterated over different cost complexity parameter choices, running different decision trees until we found signs of reduced overfitting.

This cost complexity parameter is set in order to control the deepness of the tree (number of layers of nodes) as well as the minimum number of the sub-sample size. Hence setting a positive cost complexity avoids building pure subsets which would often imply significant deepness and overfitting. Another way to see it is that setting a positive cost complexity (cp) parameter prevents the tree construction from continuing unless it reduces the overall lack of fit by a factor of cp. Hence the higher the cp parameter the more branches are pruned.

In our final Decision tree model (with a reduced set of variables), we selected a low cp parameter of 0.003 that was sufficient to overcome overfitting. We scored an accuracy of 77.68% on the training set and 77.53% on the test set, meaning that our model could generalize fairly well.

Part 3: Grouped analysis

After performing an in-depth descriptive analysis of individual observations in the personal analysis part, we capitalize on the geographical data available, to perform some grouped analysis. This grouped analysis is structured in two ways:

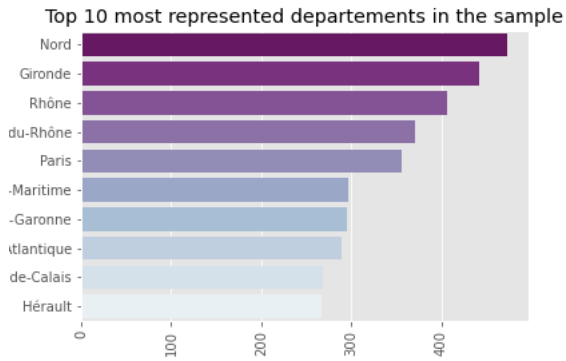
- Statistical and descriptive analysis
- Geographical analysis (Choropleth maps with folium)

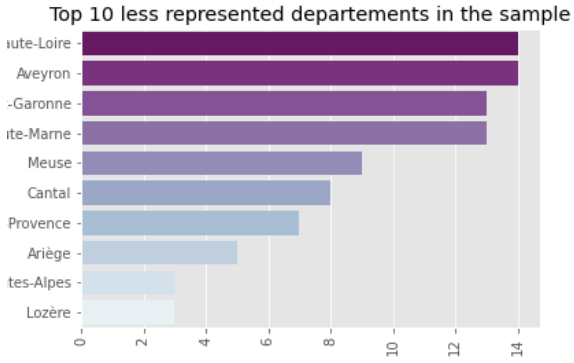
In both of those parts, three approach are taken:

- Departmental analysis
- Regional analysis
- North/South Analysis

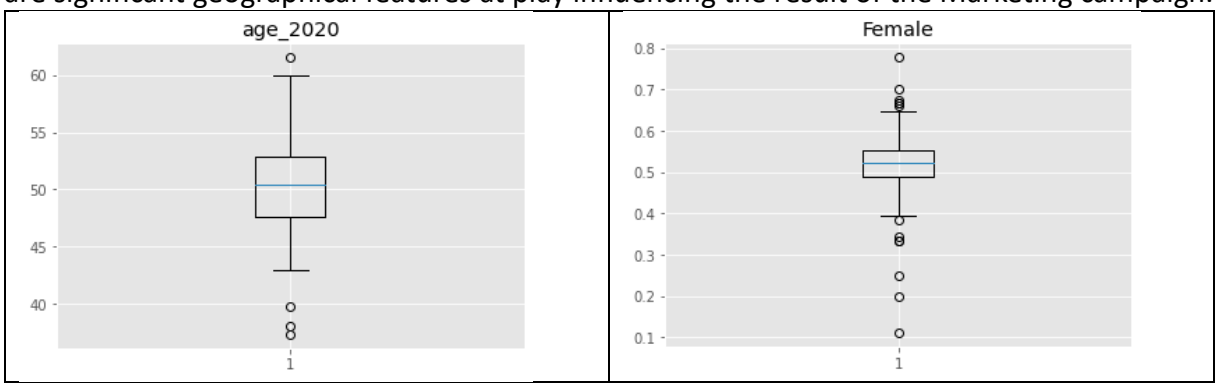
1) <u>Departmental analysis:</u>

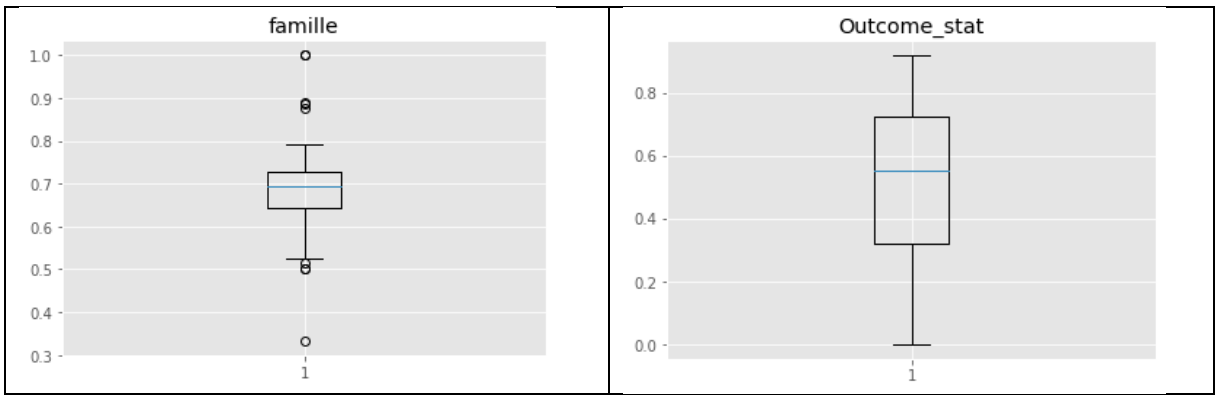
Grouping observations by department, allowed to have a precise overview of the dataset's distribution, as well as assess the level of geographical distribution heterogeneity among variables. We notice quite an uneven distribution of observations across departments.



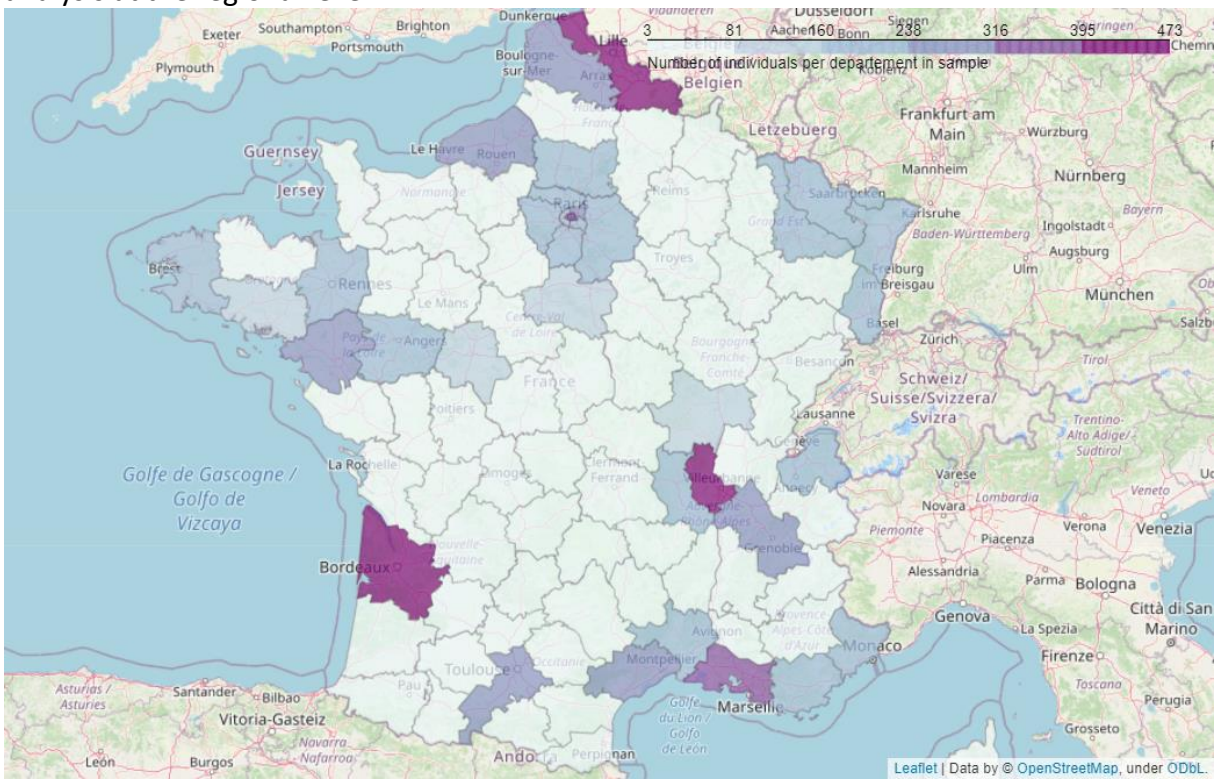


Variables distribution are computed across departments using box plots. Box plots display no significant distribution variation across departments for most variables, except the 'Outcome' variable which describes the result of the Marketing campaign. This seems indicating that there are significant geographical features at play influencing the result of the Marketing campaign.





However, the scarcity of observations in many departments prevents to build robust statistical inferences about the variables at hand (confidence intervals computed would end up being often very large, because of how poorly represented some departments are in dataset). This is made clearly apparent through a basic geographical analysis. This motivated us to pursue the grouped analysis at the regional level.

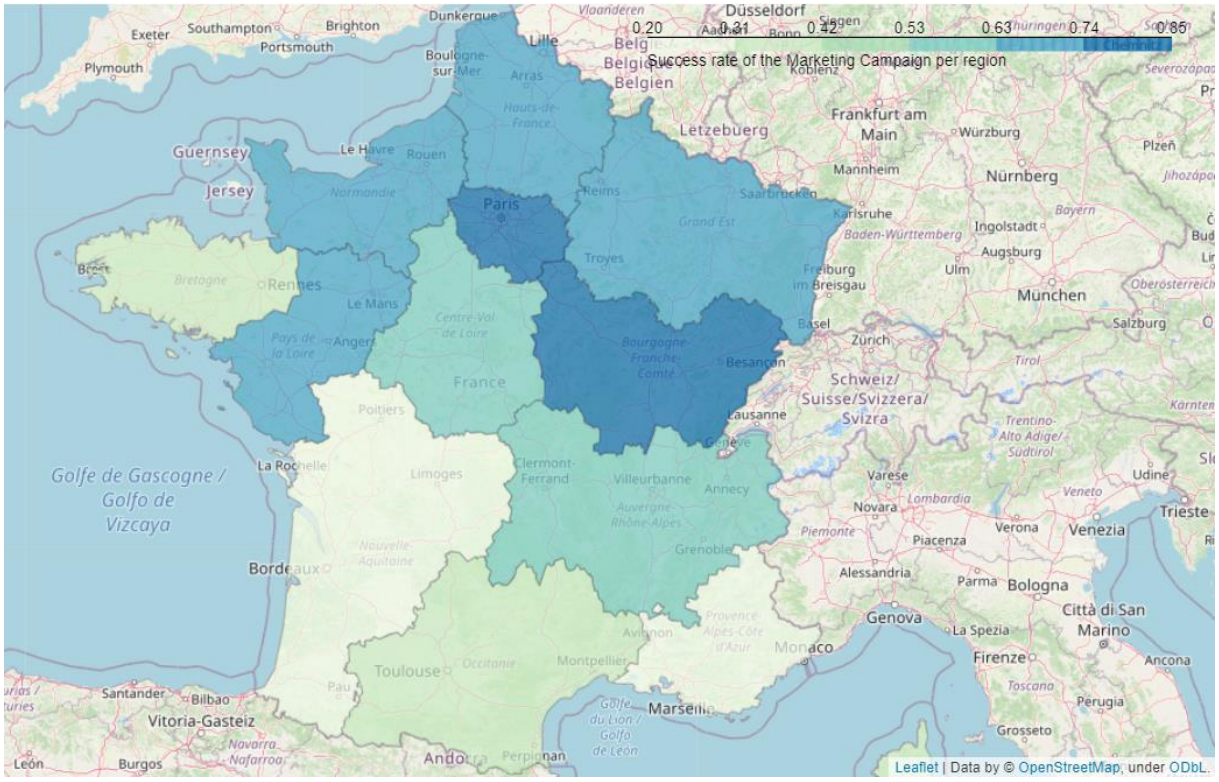


Map 1: <u>Number of individuals in the sample per department</u>

2) <u>Regional analysis:</u>

Operating at a regional level, allowed us to perform some statistically robust analysis. Grouping individuals per region confirmed the fact that the Marketing campaign success rate highly differs across geographical localization. A quick analysis of the distribution of other explanatory variables (age, activity …) reveals that this outcome heterogeneity seems to strongly rely on geographical factors.

Geographical analysis trough the implementation of Choropleth maps with folium, provides even more radical results about the geographical heterogeneity of the success rate of the marketing campaign.



Map 2: <u>Success of the Marketing campaign per region</u> – We see a clear north / south delimitation. The success rate in northern regions seems significantly higher than in the southern regions.

Out of this geographical analysis a clear delimitation between southern and northern regions stands out. This motivates a last shift in our grouped analysis: looking at the dataset separating individuals located in the northern regions, from observations located in the southern regions of France.

3) <u>North/South Analysis:</u>

Using a list of southern regions, we created a new variable 'is_south' (Boolean 1: if the observation belongs to a south region of France, 0 if not) to perform a grouped analysis based on a north/south delimitation. The geographical observation made through the regional map, was confirmed by robust statistical findings:

We observe that the average success rate of the market campaign in southern regions of France is at a low 34% while the average success rate in northern regions of France skyrockets at 72%. This striking result is confirmed by a t-test which points out a significant difference of average success between the northern and southern regions in our sample. There seem to be a strong correlation between the North/South localization and the outcome of the Marketing campaign.

The question is can we infer causation out of this relation? As a result, we analyzed the distribution of other significant variables across north and south of France, to see if they could be any hidden correlated variable able to explain such a strong relation between north/south localization and the marketing campaign outcome. Additionally, we looked at the typical profile of individuals which positively responded to the marketing campaign across the north and the south, by performing modal and median computations. It appeared that the typical profile was quite similar across northern and southern regions confirming the absence of any hidden correlation. Hence, we were confident in the predictive power of north/south grouping in better identifying the outcome of the marketing campaign. We therefore decided to add the 'is_south' variable to the predictive algorithm we created in the predictive section.

Note: Geographical analysis with folium.

In this part we implemented a geographical analysis using folium a python geo-visualization package. This package is not provided natively with anaconda, hence, to make to jupyter notebook work, you simply need to run the *environment.yml* file in your Anaconda prompt (`conda env create --name` **`your_env_name`** `--file=environment.yml`**:** this manipulation will also install any python library not natively provided with anaconda, used in this project).

We used the folium documentation (https://python-visualization.github.io/folium/quickstart.html) to implement regional and departmental Choropleth maps. This required us to use geojson data for departments and regions in France in order to be able to trace the overlays on the maps. Those geojson data were find on the following website: https://france-geojson.gregoiredavid.fr. This geographical analysis was very beneficial as it enabled us to quickly visualize striking geographical patterns in the dataset.