

**CASTRO NELSON**   **Année scolaire 2020 - 2021**

**Numéro étudiant : 11906643**

**MASTER 2 MoSEF DATA SCIENCE**

###### **RAPPORT D'ALTERNANCE**



**Méthodes statistiques appliquées à l’optimisation de la chaîne de valeur des Opérations**

**Tutrice**  **: Imane LOUKAH**

**Direction des Opérations – Levallois-Perret**

**Tuteur pédagogique** **: Rania HENTATI KAFFEL**

**CONFIDENTIEL**

**Remerciements**

Je tiens tout d’abord à remercier ma tutrice, Imane LOUKAH, et mon manager, Ghassan ABDINE, pour m’avoir fait confiance et intégrer dans l’équipe Ops Analytics de la direction des Opérations au sein des fonctions centrales de BNP Paribas Personal Finance. Merci pour leur accompagnement et leur oreille attentive face aux difficultés que peut représenter ces premiers pas dans le monde professionnel.

Ma reconnaissance va également à mes collaborateurs de l’équipe Ops Analytics : Imane BENCHARA, Sylvain BOITEUX, Emilie CHHEAN, Guillaume CLEMENT et Erwan MASSE. Merci à eux d’avoir participé à rendre cette année, pourtant parfois fort intensive, encore un peu plus agréable grâce à leur bienveillance et leurs bons conseils.

Par ailleurs, je remercie également l’intégralité de l’équipe pédagogique de la mention Econométrie-Statistiques de l’université Paris 1 Panthéon-Sorbonne pour leurs suivis, conseils et enseignements tout au long de ces deux années et plus particulièrement à Rania HENTATI KAFFEL, Marc Arthur DIAYE et Philippe DE PERETTI.

# Introduction

Ce rapport présente la synthèse du travail que j’ai effectué lors de ma formation en alternance en deuxième année de Master Modélisation Statistique, économique et financière au sein de la mention Econométrie, Statique de l’université Paris 1 Panthéon-Sobonne. Mon alternance s’est déroulée au sein de BNP Paribas Personal Finance, filiale à 100% du groupe BNP Paribas où j’ai intégré l’équipe Ops Analytics de la direction des opérations internationales, autrement dit, fonctions centrales, en tant que Data Scientist.

Ma mission était de réaliser des analyses et des modélisations à travers les différents projets sur lesquels j’ai travaillé, afin de mettre en place des processus opérationnels dans le but d’améliorer les efficacités sur l’ensemble de la chaîne de valeur des opérations. Mon travail s’est articulé autour des objectifs suivants :

— Participer à la réalisation et à la mise en production des études de data science

— Tester et comparer différents algorithmes, modèles, analyser les performances

— Travailler sur des projets de R&D en machine learning avec application métier

— Restituer les résultats à des collaborateurs aux profils opérationnels et non techniques ainsi qu’aux interlocuteurs des filiales

— Créer des fonctions pour automatiser certaines tâches méthodologiques servant à l’ensemble des data scientists de BNP PF monde.

Ce rapport est composé de deux chapitres. Après une présentation de l’entreprise qui m’a accueillie durant cette année, je reviendrai sur les projets sur lesquels j’ai travaillé. Enfin, je conclurai en parlant de ce que cette année d’alternance m’a apporté.

Table des matières

[Introduction 3](#_Toc1088792999)

[I. BNP Paribas Personal Finance 6](#_Toc2146859923)

[1. Présentation 6](#_Toc125142270)

[2. La Direction des Opérations 6](#_Toc815390114)

[3. L’entité Recouvrement et Contentieux 7](#_Toc1417140939)

[4. L’équipe Ops Analytics 7](#_Toc374028013)

[II. Présentation du Recouvrement et Contentieux 8](#_Toc937263313)

[1. Les objectifs du recouvrement et du contentieux 8](#_Toc1518757951)

[2. L’impayé 8](#_Toc1370213142)

[3. La chaîne de valeur dans les deux types de gestion 9](#_Toc273601251)

[TOPIC MODELLING: WHATSAPP CHATBOT (BRESIL) 11](#_Toc228331521)

[I. Nettoyage des données 12](#_Toc514247527)

[1. Les fondamentaux 12](#_Toc592371791)

[2. Méthode de suppression des stopwords 14](#_Toc1328369733)

[II. Stratégie de modélisation : Contrainte sur la taille des messages 17](#_Toc372916506)

[1. N-grams : approche naïve 17](#_Toc178880568)

[2. GSDMM model 18](#_Toc1917598400)

[III. Résultats finaux 21](#_Toc1547342624)

[SCORING BACK TO PAYMENT: Moratoire (Hongrie) 26](#_Toc2089363970)

[I. Variable cible : Détermination et traitement 27](#_Toc815911222)

[1. Vision uni-mensuelle. 27](#_Toc1116501645)

[2. Enrichissement : Vision dynamique. 29](#_Toc259090096)

[II. Feature Engineering & Nettoyage de données 30](#_Toc721455068)

[1. Mise au format client 31](#_Toc408741109)

[2. Nettoyage de données 32](#_Toc1847064125)

[III. Modélisation 36](#_Toc288758476)

[1. Sélection de variable et boosting 36](#_Toc382454606)

[2. Interprétation des résultats : Point de vue opérationnel 38](#_Toc1287072425)

[DAILY COLLECTION: REFONTE PYTHON (Centrale) 43](#_Toc497010514)

[I. Programme Daily Collection Report 43](#_Toc2008112108)

[1. Input et création de la base de données 44](#_Toc878008137)

[2. Fenêtre Tkinter & User experience 46](#_Toc1006032249)

[II. Génération de reportings et graphes 48](#_Toc1433203388)

[III. Environnement technique du programme. 50](#_Toc408652076)

[CONCLUSION GLOBALE 51](#_Toc266391628)

[Annexe 53](#_Toc393091042)

L’environnement du stage

I. BNP Paribas Personal Finance

### 1. Présentation

BNP Paribas Personal Finance est une filiale à 100% du groupe BNP Paribas, connue en France sous la marque Cetelem. Fin 2007, Cetelem fusionne avec l’UCB pour devenir "BNP Paribas Personal Finance" et couvre depuis tous les aspects du crédit :

— La consommation (partenariats : Carrefour, But, Conforama, IKEA...)

— L’automobile (partenariats : Peugeot, Toyota, Kia... + les autres entités du groupe)

— La distribution (partenariats : Carrefour, But, Conforama, IKEA...)

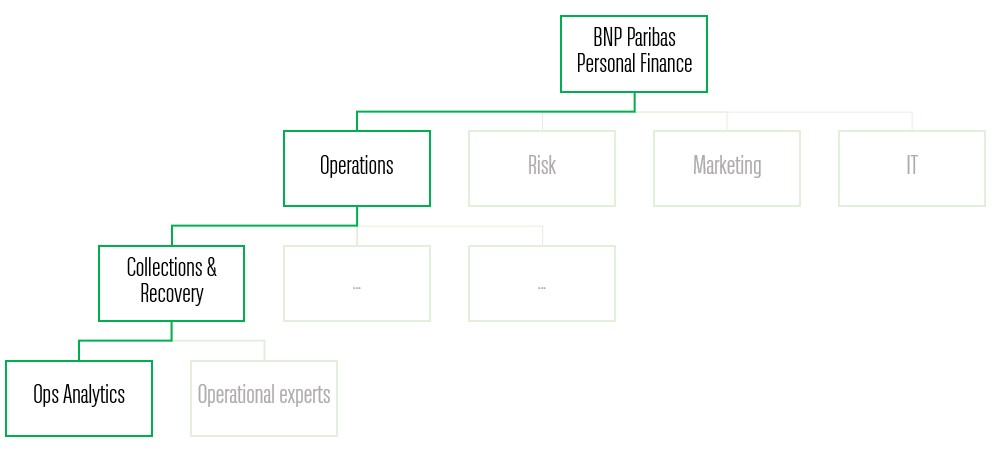
— L’immobilier

J’ai réalisé mon alternance dans l’équipe Ops Analytics au sein de l’équipe Recouvrement & Contentieux qui appartient à la Direction des Opérations Centrale de BNP Paribas Personal Finance (PF). L’organigramme de la figure 1.1 résume l’organisation de l’entreprise et montre où se situe l’équipe dans celle-ci.

### 2. La Direction des Opérations

La direction des opérations internationale (DO) est composée de 9 directions d’experts en différents domaines. La DO se charge d’impulser et de suivre la stratégie opérationnelle de l’entreprise auprès de l’ensemble des 30 filiales à l’international. C’est une direction au cœur du fonctionnement de l’ensemble des activités au sein de PF.

Ses activités ont pour but d’améliorer l’expérience client tout en optimisant les coûts, mais également de participer au développement de l’activité de PF en anticipant les évolutions du marché. Toutes ces activités peuvent se résumer en une phrase : La direction des opérations a pour objectif de placer les ressources au bon endroit et au bon moment tout en optimisant l’efficacité opérationnelle.



3. L’entité Recouvrement et Contentieux

L’équipe du Recouvrement et Contentieux fait partie d’une des 9 directions de la Direction des Opérations. Elle est composée de 4 équipes dont 4 experts opérationnels du domaine de recouvrement et contentieux ainsi que 7 experts en data science et reporting. Les experts opérationnels se chargent de :

— Surveiller les performances des équipes recouvrement des différents pays (avec la collaboration de la Direction du Risk)

— Créer et déployer les stratégies opérationnelles avec les équipes locales (ex : Gestion des impayés, lancement des projets en Data Science avec l’équipe Ops Analytics etc.)

— Assurer la mise en production des projets ou des plans stratégiques dans les filiales et suivre les évolutions

L’équipe des experts opérationnels travaille avec d’un côté les responsables du recouvrement des différents pays pour les problématiques métiers et de l’autre les data scientists pour les problématiques techniques.

4. L’équipe Ops Analytics

L’équipe Ops Analytics est une équipe relativement récente puisqu’elle a été créée fin 2016 et est composée de 4 experts de données (data analysts/data scientists) en CDI, 2 alternants et 2 stagiaires. La mission principale de cette équipe est principalement de produire les projets de type segmentation ou de scoring qui répondent aux besoins opérationnels des filiales sur le sujet de recouvrement.

Aujourd’hui, les data scientists travaillent également sur les sujets transverses au sein de la Direction des Opérations, toujours dans le but d’optimiser les efficacités et les coûts opérationnels. Les data scientists utilisent des outils comme Python, en grande majorité, y compris SAS et Tableau de façon plus secondaire. Tous les projets sont co-opérés par un trinôme (Expert opérationnel, Data Scientist, un Data Analyst de la filiale concernée) pour assurer le meilleur fonctionnement.

II. Présentation du Recouvrement et Contentieux

1. Les objectifs du recouvrement et du contentieux

L’objectif principal du recouvrement, appelé recouvrement amiable (RA), est de remettre en encours sain les clients se trouvant en impayé et ce, le plus rapidement possible en régularisant leur situation de façon pérenne, tout en assurant la relation commerciale avec le client ainsi que la maîtrise de la charge du risque. Le contentieux est défini par la déchéance du terme, à l’issue de la période de recouvrement. Elle se caractérise par :

— L’exigence de remboursement, non plus des seuls retards, mais du montant total de la créance

— L’évolution du rapport avec le client : passage d’une relation commerciale à une relation débiteur – créancier

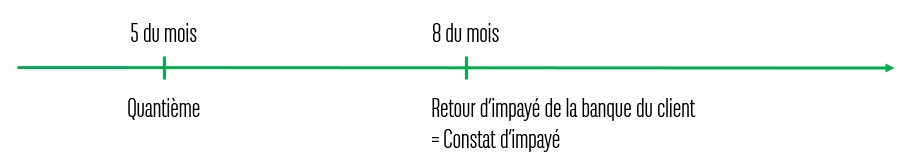
— L’entrée des engagements de procédure judiciaire

L’enjeu principal du recouvrement et contentieux est lié à deux choses qui sont tout d’abord le coût du risque c’est-à-dire le coût de provisionnement pour les dossiers risqués et que le dossier parte en perte mais aussi le coût de gestion des dossiers impayés. La stratégie au sein du recouvrement et contentieux est donc déterminée en fonction de la performance des gestionnaires des impayés, ce que l’on appelle l’efficacité opérationnelle.

2. L’impayé

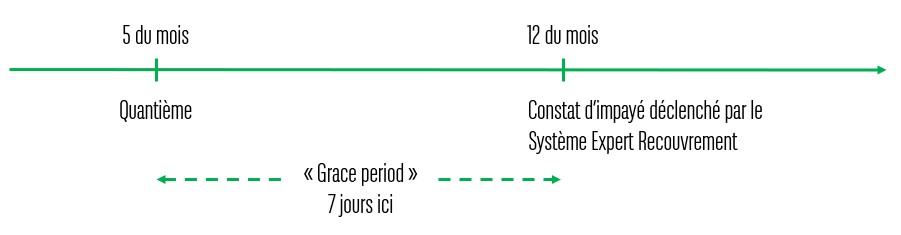
Chaque pays a ses propres stratégies de gestion des impayés, qui sont adaptés à l’environnement culturel, social et économique du pays. Il existe deux types de constat d’impayé.

**Dossiers prélevés (MSO, MSOA)**



Dans le cas d’un dossier prélevé, la banque du client va recevoir une première demande de prélèvement sur le compte du client le 5 de chaque mois. En cas d’échec de prélèvement, le Système Expert va recevoir un retour d’impayé et va relancer automatiquement une deuxième demande de prélèvement (MSO, MSOA). Si la mensualité n’a toujours pas été payée, un constat du premier impayé est enregistré dans le Système Expert.

**Dossiers non prélevés (payement par chèque, mandat postal, cash, internet...)**



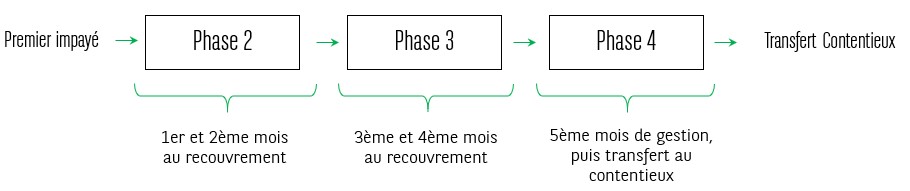
Dans le cas d’un dossier non éligible au prélèvement, la "due date" est également le 5 du mois par exemple. Seulement, si la mensualité n’a pas été payée à temps, le Système Expert va laisser 7 jours pour que le client régularise sa situation de manière autonome avant d’enregistrer un retour d’impayé dans le système.

3. La chaîne de valeur dans les deux types de gestion

Il existe deux façons de gérer les impayés dans les 30 filiales de BNP Paribas Personal Finance : la gestion par flux (France, Espagne...) et la gestion par bucket (Chine, Afrique du sud...).

**Gestion par flux/phase**

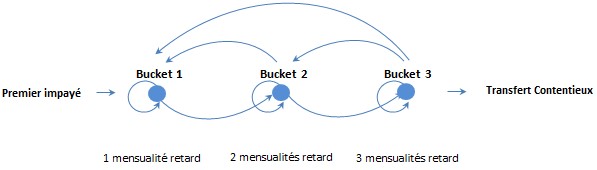
Chaque dossier est traité en fonction du temps qu’il a passé au recouvrement.



Dans un process de type Phase, les dossiers peuvent soit être régularisés donc sortir du recouvrement, soit être non-régularisés et donc avancer dans la phase suivante.

**Gestion par bucket**

Chaque dossier est traité en fonction de son nombre de mensualités en retard



Dans le cas d’un process de type Bucket, le dossier peut :

— Avancer dans le bucket suivant s’il y a 1 mensualité de retard de plus

— Reculer dans les buckets précédents s’il régularise au moins une mensualité

— Rester dans le même bucket si le nombre de retard ne change pas

— Sortir du recouvrement (retour à l’encours sain) s’il régularise tout

Dans ces deux types de process, nous retrouvons les mêmes moyens de gestion. Les gestionnaires de relation client peuvent, selon le montant total de crédit ou selon le niveau de risque du client, envoyer un mail ou un SMS, laisser un message vocal, appeler directement le client ou outsourcer le traitement.

Projets Data Science & Reporting

TOPIC MODELLING: WHATSAPP CHATBOT (BRESIL)

Il s’agit du premier projet que j’ai eu à réaliser lors de mon Alternance. Il s’agissait d’une étude d'analyse textuelle portant sur des messages Whatsapp. Cette analyse comportait deux particularités :

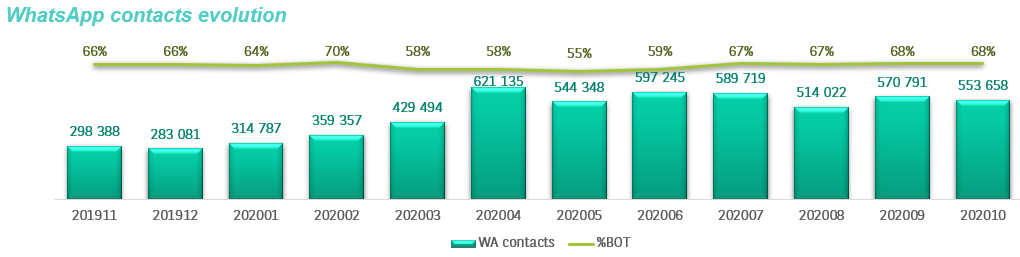
* Les messages étaient envoyés par des clients brésiliens au Service après-vente de Cetelem Brésil, ils étaient donc rédigés en portugais.
* Le format Whatsapp couplé à un système de Chatbot, les messages étaient souvent très courts, rendant difficile l’application des méthodes traditionnelles de nettoyage de données et de modélisation.

***Genèse du projet***

Ce projet, bien qu’ayant réellement débuté en Octobre 2020 a été impulsé plutôt durant l’été de la même année.

À la suite de la première vague quasi-mondiale de COVID-19 (Autrement dit, à partir d’Avril/Mai 2020), les services de Cetelem Brésil ont vu une explosion du volume des contacts reçus. Cette hausse a été plutôt bien encaissée par le Chatbot (voir graphe ci-dessous). En effet, le pourcentage de demandes traitées par ce dernier n’a pas excessivement diminué en proportion, cependant comme le volume de contact ayant augmenté sur la période, il y a eu une recrudescence d’appels aux conseillers Cetelem Brésil.

De plus, on constate que même en proportion le Chatbot a été moins efficace au cœur de la crise sanitaire.



L’objectif était donc le suivant : Déterminer les principales raisons de contact que le Chatbot ne parvient pas à traiter efficacement et qui mène à un appel direct aux conseillers téléphoniques de Cetelem Brésil. Le traitement de ce sujet d’étude étant une première, une réflexion importante a dû être faite concernant la méthodologie afin de favoriser le re-use à d’autres projets portant sur l’analyse de données textuelle.

***Présentation de la base de données***

La base de données qui nous a été fournie contenant quelques dix millions d’observations contenant les conversations ayant eu lieu entre Juin et Août 2020, toujours entre les clients et le service après-vente de Cetelem Brésil.

La base contenant finalement assez peu de variables : Les messages Whatsapp en eux-mêmes, un label indiquant si le message a été traité par le Bot ou non, le sujet du contact, l’incrémentation du nombre de messages échangés…

Pour résumer, une ligne correspondait à un message envoyé par le client au service après-vente. Chaque message supplémentaire venait enrichir la base d’une observation supplémentaire, la variable d’incrémentation suscitée permettant de tenir le compte.

I. Nettoyage des données

Cette partie a été la plus importante, c’est ici que la réflexion sur la méthodologie s’est concentrée.

Il a fallu trouver un équilibre entre le nettoyage provenant de :

* La connaissance du portugais. Nous avons minimisé le curseur ici car c’est ce qui fera que notre méthodologie pourra être facilement appliqué à d’autres langues.
* La connaissance de la base de données elle-même et du « format » Whatsapp
* Les outils algorithmiques de nettoyage des données

1. Les fondamentaux

*Expressions régulières*

* Supressions des mots mélangeant des chiffres et des lettres (*Exemple: h4ck3r*).
* Suppression URLs avec http ou www
* Suppression des emails et autres mentions utilisant le sigle @
* Suppression de mots trop longs (plus de 18 caractères) et trop courts (moins de 2 lettres)
* Suppression de la punctuation
* Traitement des messages trop courts pour contenir de l’information, ici on supprime donc les messages contenant moins de 2 mots. Ce filtre est crucial car permet de traiter les messages ne contenant qu’une formule de politesse type « *Ola* » ou encore plus important les phrases sans espaces, qui sont malheureusement monnaie courante sur un canal de communication tel que Whatsapp, telles que **«***Olasogabrieleuprecisodeverafatura* » qui revenaient assez régulièrement.

*Re-use: Stopwords provenant d’anciens projets*

Le traitement des stopwords a été l’étape de nettoyage sur laquelle que je me suis le plus concentré. Des listes de Stopwords, provenant d’anciens projets et des collaborateurs brésiliens eux-mêmes, existaient déjà et ont donc pu être intégré au script assez facilement. On a un exemple ci-dessous des mots/expressions concernés :

**Stopwords** des collaborateurs au Brésil

**Common expressions**:

[oi,

bom dia,

boa tarde,

ola,

ok,

sim,

boa noite…]

**Stopwords** d’anciens projets NLP

**Common words**:

[a,

ainda,

alem,

ambos,

antes,

apos,

aquele,

muita…]

Quand bien même ces listes étaient plus longues que ce que cet échantillon laisse apparaître, il était nécessaire d’accentuer le nettoyage. En effet, on imagine aisément que sur plusieurs millions de messages, d’autant plus écrits à la main, il n’y a pas que les expressions communes qui peuvent poser un problème. Parmi eux : Les fautes d’orthographe, les mots inexistants, les abréviations, les anglicismes…

2. Méthode de suppression des stopwords

À la suite de cette première étape de nettoyage, j’ai utilisé une méthode combinant deux outils algorithmiques différentes :

* La librairie FastText développée et maintenue par Facebook.
* L’algorithme de classification non-supervisée des K-means.

*Librairie FastText*

FastText est donc une librarie open-source mise à disposition par les équipes IA de Facebook. L’idée derrière cet algorithme est que la structure morphologique d’un mot contient l’information principale concernant le sens du mot que l’on souhaite étudier.

Ainsi, FastText s’essaie à traiter chaque mot comme l’agrégation d’un ensemble de syllabes grâce au Word Embedding. Le word embedding désigne un ensemble de techniques de machine learning qui visent à représenter les mots ou les phrases d’un texte par des vecteurs de nombres réels, décrits dans un modèle vectoriel (ou Vector Space Model). Ces nouvelles représentations de données textuelles ont permis d’améliorer les performances des méthodes de traitement automatique des langues (ou Natural Language Processing), comme l'Analyse de Sentiment ou le Topic Modelling. C’est l’application à ce dernier champ d’étude qui nous intéresse ici.

Cette méthodologie est appliquée sur une dizaine de langues différentes et donne lieu à un modèle pré-entraîné permettant d’effectuer la transformation des mots en vecteurs. Dans mon cas, j’ai évidemment utilisé sa version portugaise.

Dans notre use case, il a permis d’atteindre deux objectifs :

* Il a permis de filtrer et supprimer des mots qui n’existaient pas. Un échantillon ci-dessous :
* *['gacpe', 'quwro', 'vezrs', '\_bom', 'favpr', 'owue', 'webx', 'giw', 'veuo', 'opsao', 'dfui', 'eztou', 'noeu', 'swnha', 'vvs', 'essw', 'ezta', 'shptme', 'sqber', 'fszer', 'akico', 'rjcep', 'rufb', 'ifd', 'snh', 'errp', 'adzp', 'vslor', 'vct', 'quwr', 'gcr', 'mgcep', 'usa\_lo', '\_on', 'bolwto', 'vrjo', 'qnn', 'msu', 'bdn', 'atw', 'wwlllww', 'oyo', 'vsav', 'twnho', 'gavlak', 'fizno']*
* Il a également permis de vectoriser nos messages. Chaque mot d’un message est donc projeté dans un hyperplan, c’est ainsi que ces derniers ont été convertis en valeur numérique nous permettant de passer à la seconde étape de notre processus : Les K-means.

*K-means clustering*

On peut maintenant parler de l’algorithme des K-means, principal outil utilisé dans la construction de nouvelles listes de stopwords. Il s’agit donc d’un algorithme d’apprentissage non-supervisé, c’est-à-dire qu’on va s’en servir pour grouper des individus « non-labelés » (Il faut entendre ici qu’on n’a pas de variable cible discriminante) dans différents « clusters ».

*Rapide rappel du fonctionnement des K-means* :

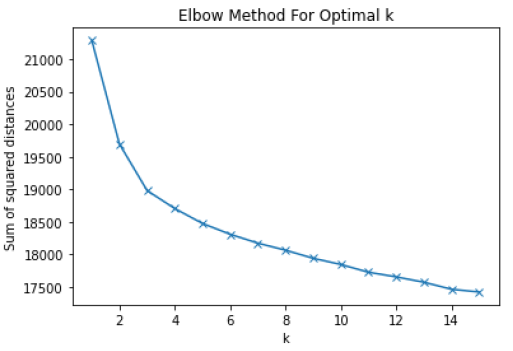
En gardant le reste de côté, le « K » définit le nombre de clusters pré-définis par nos soins (au moment de lancer le modèle) qui devront être « remplis » dans le processus de clustering. Pour K = 2, il y aura deux clusters, pour K = 3, trois clusters, et ainsi de suite.

Il s’agit également d’un algorithme à caractère récursif. Les clusters vont dans un premier temps être construit à partir de K points aléatoires (appelés centroïdes) auxquels vont venir se greffer les points restants en fonction de la distance qui les sépare : en l’occurrence on parle de distance euclidienne.

Donc récursivement ensuite l’algorithme va calculer le centre de ces « nuages » de points et procéder à nouveau à des groupements fondés sur cette même distance euclidienne. Ces ajustements et groupements continueront jusqu’à ce que les clusters se stabilisent ou selon des conditions d’arrêts (par exemple, nombre d’étapes maximales pour éviter les apprentissages non convergents).

Dès lors, pour déterminer le nombre optimal de clusters K dans nos données, on utilise la « méthode du Coude ». Le graphe ci-dessous illustre le résultat de cette méthode :

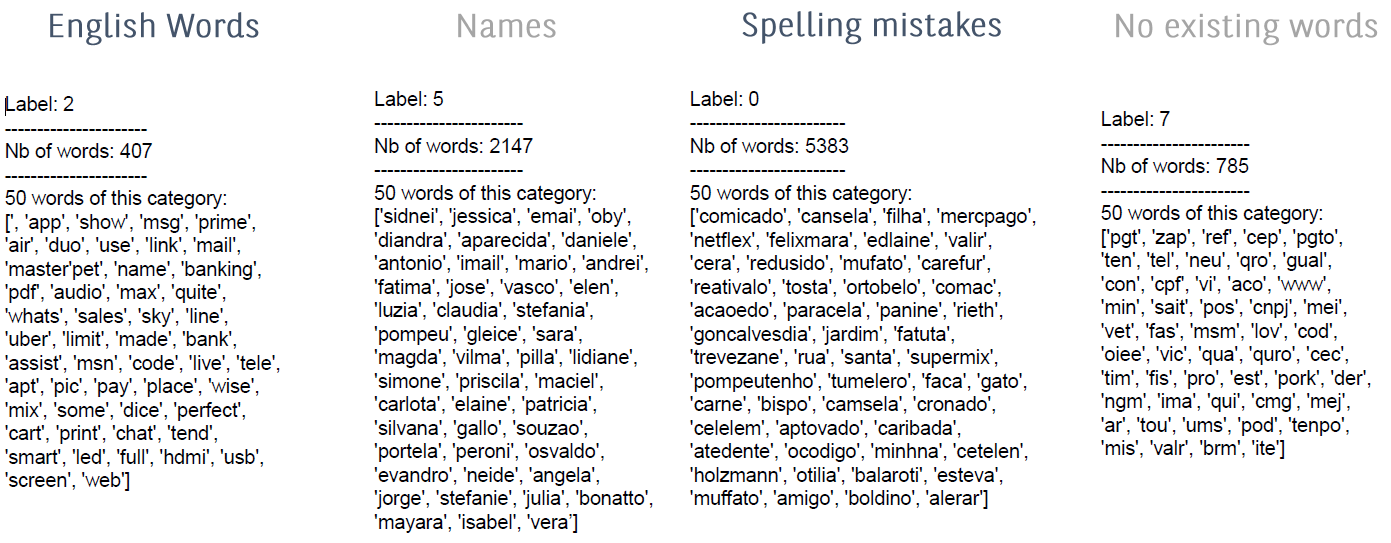
*Résultats* :



**On a choisi l’inertie pour notre critère de performance.** L’inertie est la somme des distances au carré entre les points d’un même cluster par rapport à leur centroïde (on additionne alors le résultat avec celui des autres K-1 clusters). Autrement dit, plus cette métrique est basse, mieux c’est.

* Comme son nom l’indique, la méthode du coude consiste à chercher un angle dans cette courbe.
* Ici, malheureusement aucune K valeur ne semble se détacher, au mépris de groupes trop grands qui seraient difficiles à interpréter on a donc choisi k = 8 car c’est à partir de là selon moi que la courbe commence à s’aplatir plus légèrement.

Ci-dessous un échantillon parmi les résultats les plus significatifs et interprétables que nous avons obtenu grâce aux 8 clusters :



On constate donc que l’algorithme est efficace pour grouper des mots en fonction de leur structure grammaticale.

* Le premier cluster semble davantage contenir des mots anglais voire même des marques, ça n’apparaît pas dans cet échantillon mais « Netflix » et « Amazon » ressortaient de ce cluster par exemple.
* Le deuxième cluster, avec plus de mots différents, contient des noms et prénoms.
* Le troisième nécessite quelques connaissances de la langue mais il s’agit de fautes d’orthographe
* Le dernier contient généralement des mots très courts sans aucun sens.

*Nouveaux stopwords & routine d’utilisation*

Suite à cette partie portant sur le nettoyage des données, les mots présents dans les clusters présentés sont venus enrichir des listes de Stopwords qui ont ensuite été sauvegardées en format .pkl. Facile d’utilisation, il permet de faire appel à ces listes au début du script avec Python (type « list »).

L’idée est donc de ne pas avoir à relancer FastTest et les K-means à chaque utilisation du programme. Deux raisons à cela :

* FastText est assez long à charger en mémoire
* Les résultats K-means peuvent légèrement changer d’une relance à l’autre mais surtout, et ce en raison de l’aléatoire au moment de l’initialisation de l’algorithme, le cluster n°1 ne sera pas toujours le cluster contenant les mots en anglais par exemple. Ce qui rend impossible toute automatisation de cette partie du processus.

Les listes ainsi sauvegardées sont ainsi appelées chaque fois en début de script afin de nettoyer la base de données.

II. Stratégie de modélisation : Contrainte sur la taille des messages

Après le nettoyage des données, il était temps de choisir le meilleur modèle afin d’extraire les principales raisons de contact non traitées par le Bot.

A cette étape du processus, nous avions encore plusieurs choix pour répondre à la problématique :

* Utilisation de statistiques descriptives afin d’extraire des informations « générales » sur la base de données. Le problème étant qu’une solution manuelle de ce type était déjà utilisée sur Excel, il est d’ailleurs difficile de ne se concentrer que sur des méthodes de comptages de ce type pour extraire des sujets plutôt que des mots.
* Une solution alternative aurait été l’apprentissage supervisé, mais pour des raisons déjà évoquées, ce n’était pas possible. Il y avait bien une variable indiquant quel était le sujet du contact, mais celle-ci semblait remplie manuellement et possédait de toute manière plus d’une centaine de modalités différentes, rendant impossible la classification.

On comprend donc qu’il nous fallait une stratégie généralisable, automatisable et ne nécessitant pas de variable cible.

Nous nous sommes donc essayés à deux méthodes différentes :

* Une approche naïve fondée sur les N-grams
* Algorithme de topic modeling non supervisé

1. N-grams : approche naïve

On a d’abord utilisé cette approche afin de classer les messages entre différents sujets (≈Raisons de contact). Cette méthode, bien qu’encore assez manuelle, permettait d’obtenir des résultats satisfaisants : 50% des messages finissaient classés dans un thème définit par nous-même.

Pour ce faire, chaque message a été transformé en format bi-gram. Exemple de conversion :

* « querer ver fatura detalhada »
* « Querer\_ver, ver\_fatura, fatura\_detalhada ».

Cela permet ensuite de compter le nombre d’apparition de chacun de ces couples de mots afin d’extraire les N plus fréquents (disons 200, mais c’est modulable). Ce seuil de N a été fixé de façon arbitraire, une méthode plus inteligente aurait pu être définie par la suite mais cette aproche était encore en pase de test.

On a ensuite trié ces bigrams en fonction de sous-section de mots y apparaissant :

* “Sujet factures” : Regroupait les bigrams contenant les mots “fatura” et “boleto” (synonyme de facture en portugais). On a eu un raisonnement similaire pour les sujets « fraude », « cartes de crédit », « informations personnelles » etc.

On n’est au final pas allé plus loin car bien qu’intéressante, cette méthode était encore beaucoup trop manuelle et proche de la solution que possédait le Brésil sur Excel.

2. GSDMM model

Ce modèle implémente l’algorithme « Gibbs Sampling for a Dirichlet Mixture Model » développé par Yin et Wang dans un article de 2004 intitulé “A dirichlet multinomial mixture model-based approach for short text clustering”. Cet article définit le modèle de la façon suivante :

*“Imagine a professor is leading a film class. At the start of the class, the students are randomly assigned to K tables. Before class begins, the students make lists of their favorite films. The professor repeatedly reads the class role. Each time the student's name is called, the student must select a new table satisfying one or both of the following conditions:*

* *The new table has more students than the current table.*
* *The new table has students with similar lists of favorite movies.*

*By following these steps consistently, we might expect that the students eventually arrive at an "optimal" table configuration.”*

*Dès lors, pourquoi ce modèle plutôt qu’un algorithme plus traditionnel tel que le LDA ?*

L’algorithme LDA nécessite une forme de taille plancher des messages afin de performer correctement. Bien qu’il n’y ait pas de règle fixe et pré-établie, la littérature scientifique s’accorde à indiquer qu’un minimum de 50 mots par messages est nécessaire, c’est généralement le cas d’un email ou d’une conversation orale retranscrite.

Dès lors, des messages aussi courts que des tweets ou des Whatsapp ne remplissent pas les conditions adéquates. L’exemple ci-dessous aide particulièrement à le comprendre :



En dehors de la présence de ce qu’on pourrait considérer comme des stopwords, on se rend aisément compte qu’on ne pourra pas extraire d’information d’un message aussi court. Dans notre cas d’usage, après la phase de nettoyage, la taille moyenne des messages Whatsapp était d’un peu moins de 3 mots.

*Quelle est la différence entre ces deux algorithmes ?*

**Latent Dirichlet Allocation (LDA) :**

C’est un modèle qui vise à extraire le sens (entendre ici le « thème ») d’un groupe de mots en analysant les similarités entre ces derniers et les mots qui les entourent. On comprend dès lors que ce modèle ne va pas extraire un seul et unique thème d’un message, mais va au contraire considérer qu’il est composé d’une multitude de sujets. Evidemment, il ne va pas non plus souligner le topic de ce groupe de mots en lui-même, mais va plutôt chercher à mettre en évidence les groupes de mots qui semblent appartenir à un même sujet.

Le modèle LDA donc suppose que chaque document est un mélange d’un petit nombre de sujets ou thèmes et que la génération de chaque occurrence d’un mot est attribuable (probabilité) à l’un des thèmes du document. Donc ce n’est pas le thème qui a une probabilité d’exister grâce à un mot, c’est le mot qui a une probabilité d’être généré par le sujet.

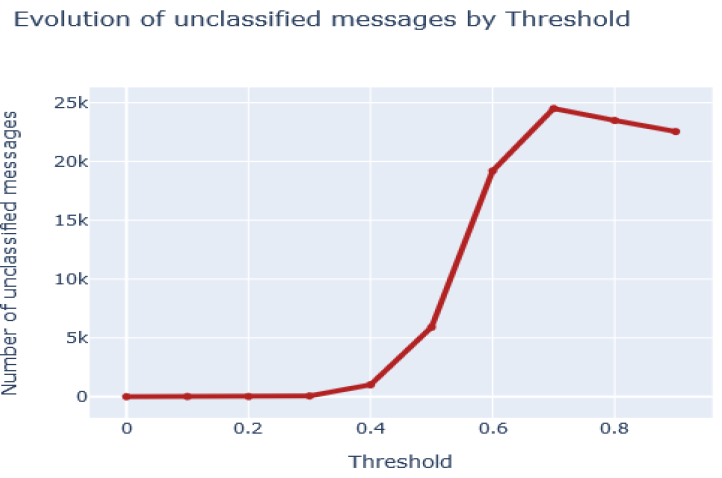
**GSDMM :**

Au contraire, pour les GSDMM, un document = un sujet. Alors que LDA donne la probabilité d’un document d’appartenir aux différents clusters existants (Un document étant alors composé, comme nous l’avons dit, de plusieurs sujets mais avec un thème dominant tout de même), les GSDMM font l’hypothèse que les mots contenus par le document sont générés selon le même sujet.

Cette probabilité que les mots du document soient générés par X ou Y cluster fait d’ailleurs partie des paramètres que l’on peut modifier afin de considérer que le document appartient ou non à ce cluster.

*Par exemple, si la probabilité du document d’appartenir au cluster X est de 0.3, on peut considérer que c’est trop peu et que le modèle n’est pas parvenu à discriminer correctement le document. Dès lors, on peut fixer à 0.4 le seuil minimum, si la probabilité est en-deçà de cette limite, le document ne sera pas classé*.

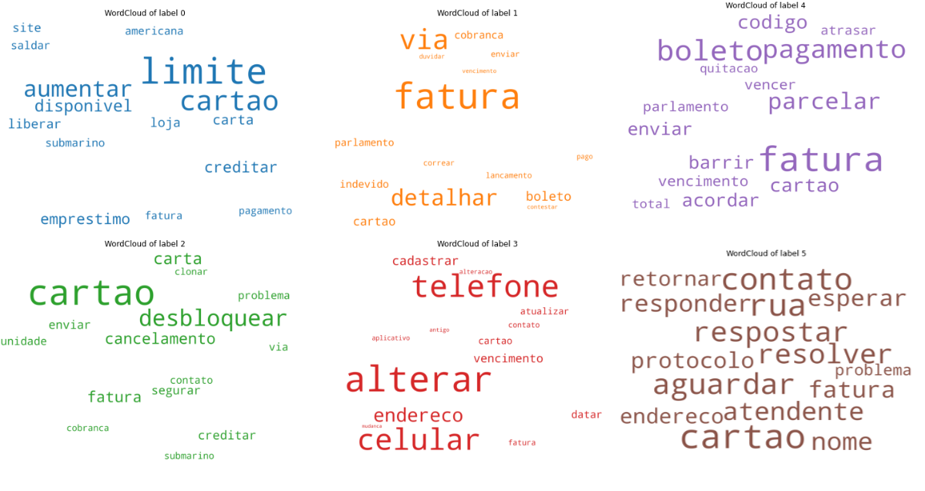
Cela peut avoir un impact considérable sur les résultats finaux comme le démontre le graphe ci-dessous :



On remarque qu’au-delà de 0.4, le nombre de messages non classés explose jusqu’à atteindre une forme type sigmoïde. Afin de garder des propriétés discriminantes mais sans pour autant perdre trop de messages en route, on s’est donc fixé sur 0.4 en guise de seuil de décision.

III. Résultats finaux

Le principal résultat des GSDMM dans notre cas d’usage sont des nuages de mots qui aideront ensuite à interpréter les clusters dans le but de leur attribuer un intitulé de sujet (et donc d’une raison de contact). Ci-dessous un exemple de ce qu’on a pu obtenir :



Dans notre cas, le choix de demander au modèle de trouver 6 clusters provient de l’expérience. Avec moins de topic on perdait en capacité discriminante, avec plus on perdait en sens avec des clusters « fourre-tout » regroupant des mots paraissant peu connectés entre eux.

Dans ce cas, voici un exemple d’interprétation que l’on pourrait attribuer à chaque cluster en fonction de ce nuage de mots :

1 : Seuil de paiement de la carte de crédit

2 : Vision Facture détaillée

3 : Problème avec la carte de crédit

4 : Modification des informations personnelles du client

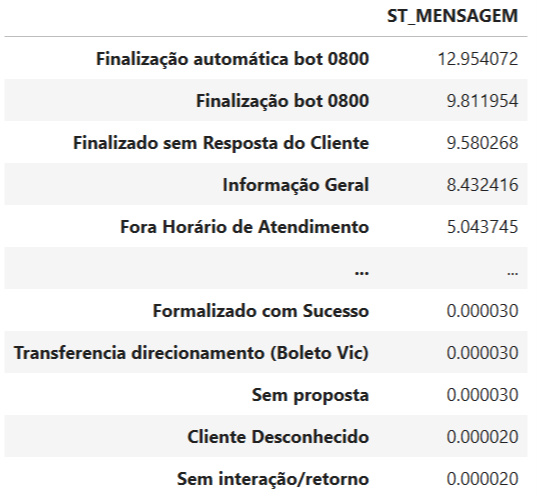
5 : Opération sur facture: Paiement, échéance, report

6 : Modification des informations personnelles du client

*Métrique de performance*

Dans le but de mesurer la performance de notre méthode dans la détermination des principales raisons de contact, j’ai créé une métrique de performance assez simple fondée sur la colonne dont les modalités indiquent les raisons de contact de chaque client. Comme déjà mentionné, celle-ci n’était pas utilisable en guise de target du fait de son nombre trop important de modalités différentes.

Ci-dessous on peut constater le déséquilibre apporté par cette variable et l’effet d’entonnoir dans le remplissage manuel du sujet du contact :



Sur 184 modalités, les 20 plus importantes représentent 80% des observations.

A partir de là, on a filtré notre base de données en fonction de cette même colonne avec l’objectif de garder les modalités reliées aux 6 clusters précédemment évoqués. Prenons le cas du problème avec la carte de crédit :

|  |  |
| --- | --- |
| **LABEL MENSAGEM** | **Proportion de label classes comme “Cartao problema tema"** |
| **retencao de cartao** | **100%** |
| **cancelamento de seguros** | **73%** |
| **desbloqueio de cartao** | **72%** |
| **desbloqueio de senha** | **70%** |
| informacao seguros | 69% |
| **reenvio de cartao** | **67%** |
| **rastreio do cartao ou recebimento da senha** | **64%** |
| **retencao seguros** | **62%** |
| **bloqueio preventivo** | **61%** |
| fatura digital | 60% |

Le tableau se lit alors comme suit : 100% des observations catégorisées comme « retencao de cartao » ont été classée dans le cluster « Cartao problema tema ». En dehors de cette modalité, on remarque qu’on orbite généralement autour de 70%, ce qui est une bonne chose. Notre modèle a bien réussi à discriminer les documents de la base de données en fonction des mots, et donc du sujet, qui y apparaissent.

Les tableaux pour les autres clusters sont présents en annexe mais ne seront pas commentés ici.

*Environnement*

|--- Pickles

| +-- abbreviations.pkl

| +-- Conjugaison.pkl

| +-- Dictionnaire\_Correction.pkl

|--- logs

| +-- Contains all “DD\_MM\_YYYY\_HH\_MM\_SS.log” files, description later.

|--- Results

| +-- **Wordclouds.png**

| +-- **TopWords.csv**

|--- Stopwords\_lists

| +-- names.txt

| +-- common\_expression.txt

| +-- Faltas.txt

| +-- EnglishWords.txt

|--- cleaning.py

|--- model.py

|--- src

| +-- cleanings\_tools.py

| +-- contants.py

| +-- libraries.py

| +-- ModelGSDMM.py

| +-- model\_tools.py

|--- requirements.txt

|--- Parameters.yaml

|--- README.md

*Mise en production*

Tout le script a donc été packagé en deux parties :

* Le script de nettoyage des données, il peut être lancé directement depuis le terminal de commande.
* Le script de modélisation.
  + Celui-ci peut également être lancé depuis le terminal, mais nécessite l’utilisation du fichier Parameters.yaml qui, comme son nom l’indique, aide à la paramétrisation du modèle (Base de données, seuil, nombre de clusters).
  + Les fichiers Yaml présentent l’avantage d’être intuitif s’agissant de sa modification par les utilisateurs du modèle sans connaissance de Python.

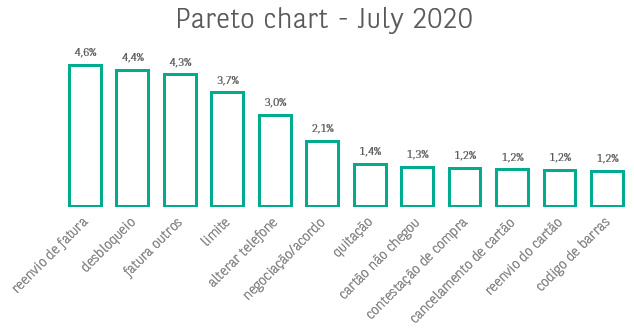
Le script produit donc 3 résultats différents :

* Les nuages de mots qui représentent chacun des clusters construits par le modèle.
* Le Top X des mots les plus importants de chaque cluster (Ce X fait également partie des paramètres à la main de l’utilisateur).
* Une base de données post-nettoyage labellisée par les prédictions du modèle. Elle peut donc être exportée par les collaborateurs brésiliens afin de procéder si nécessaire à des études transverses sur des données déjà nettoyées.

**Conclusion et usage opérationnel**

Cette solution a aidé les équipes après-vente brésiliennes à améliorer la qualité de leur prise de contact grâce à une meilleure connaissance des tendances de besoin des clients. Par exemple, nous avons remarqué qu’une grande partie de ces messages non traitées par le Chatbot concernaient la volonté du client de voir sa « facture détaillée » (voir graphe ci-dessous).

En effet, bien qu’il existait déjà une telle option dans le Chatbot, il semble que de nombreux clients ne la voyaient pas et privilégiaient le contact avec un conseiller.



Dès lors, l’une des actions mises en place par le Brésil suite à ce projet a été de déplacer le bouton « facture détaillée » en haut de la liste des interactions proposées par le Chatbot, afin qu’il soit davantage visible par les clients.

SCORING BACK TO PAYMENT: Moratoire (Hongrie)

Ce score fait suite à la décision de la Hongrie de mettre fin au moratoire qui avait été instauré au début de la crise sanitaire.

Un moratoire est un terme juridique qui désigne une décision d'accorder un délai ou une suspension volontaire d'une action. Disposition légale, nécessitée par des raisons impérieuses d'intérêt public, suspendant d'une manière générale l'exigibilité des créances, le cours d'actions en justice. Dans notre cas, il s’agissait donc d’un report des échéances dues par les clients de Personal Finance en Hongrie, et ce sans nécessité pour ce client de justifier une quelconque raison objective à cet arrêt des paiements.

Après quelques extensions du moratoire, le mois de Juin 2021 devait être celui marquant la fin du moratoire. Dès lors, une large campagne d’appels devait être mis en place afin de pousser les clients les plus réticents au retour au paiement.

***Un score à usage quasiment unique***

Comme vous l’avez sans doute compris à la lecture des précédentes phrases, ce score n’était pas amené à perdurer dans le temps, il s’agissait d’un modèle qu’on pourrait quasiment considérer comme « One-Shot », s’affranchissant d’un certain nombre de contraintes habituelles :

* Explicabilité et impact des variables explicatives
* Stabilité temporelle

Dans notre cas, il s’agissait d’un score orienté sur la performance. Il fallait obtenir les meilleures métriques possibles afin de s’assurer que notre ciblage des clients les plus à risque de ne pas reprendre le paiement soit efficace.

***Description de la base***

Les données utilisées contiennent des indicateurs, à la fois socio-démographique et de paiement, de la totalité du portefeuille du moratoire au point de vue mensuel.

Pour une partie des données portant sur le profil du client en lui-même, nous avons décidé d’utiliser leur « version » de Février 2020 au moment de l’entrée dans la crise sanitaire afin de s’assurer d’obtenir une photographie cohérente de l’état du portefeuille hongrois étant donné que nous étions au sortir de la crise au moment du développement du score. Évidemment, des discussions et doutes peuvent être amenés face à cette décision, mais les variables choisies au sein de cette base sont de toute manière assez peu variante.

I. Variable cible : Détermination et traitement

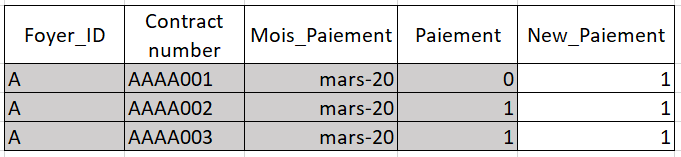
Une certaine partie du travail a été abattue sur la construction de la variable. Le but étant de tenir compte de la dynamique de paiement du client depuis le début du moratoire afin de déterminer si à la sortie il sera en mesure de payer ou non.

1. Vision uni-mensuelle.

Pour ce faire, on a utilisé les données du portefeuille du moratoire de Mars 2020 à Avril 2021. Ainsi, chaque mois, pour un client donné, nous pouvions savoir si celui-ci avait payé ou non sa mensualité.

*Clients multi-contrats*

Avant de raisonner à l’échelle multi-mensuelle, il fallait déjà trouver un moyen de traiter les clients qui possédaient plusieurs contrats. En collaboration avec les Experts Métiers, il a été décidé du traitement suivant. Pour un client A possédant trois contrats en date de Mars 2020 :



Si plus de 50% des contrats détenus par le client ont été payé pour le mois données : On considère qu’ils ont tous été payés.

If >= 50% of contracts detained by a client are paid for a given month : We consider that all of them have been paid

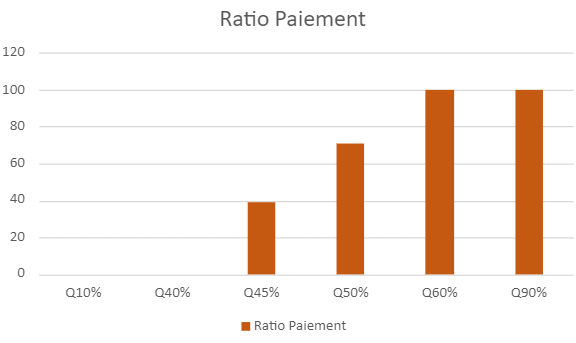
Donc dès lors qu’il a payé plus de 50% des mensualités des contrats qu’il devait, on pouvait considérer qu’il avait payé la totalité.

Encore une fois, si un tel traitement pourrait paraître aberrant sur d’autres types de projet de scoring : Score d’octroi de crédit dans un département de risque par exemple ou même un score de passage au contentieux d’un client à la suite d’un premier impayé, impliquant chacun un montant de provision qui va venir compenser les risques engagés auprès de ces contreparties. Ce n’est pas le cas ici où nous tentons avant tout de définir un profil, dès lors un client qui a payé 50% de ses contrats à un mois donné, en pleine crise sanitaire, peut être considéré comme un « bon » client et justifie ainsi de ne pas engager de ressources financières et humaines auprès de lui afin de le motiver à un retour au paiement à la fin du moratoire.

*Paiements partiels*

Un autre point est également venu poser la question de la granularité de notre classification binaire : Les paiements partiels.

Parfois, les clients payaient certes, mais ne payaient pas la totalité de la mensualité. Devait-on alors apporter une nuance particulière à la règle de décision précédemment évoquée afin de classer plus finement nos individus ? Encore une fois, cette question a été débattue au sein de l’équipe du projet. Nous avons constaté que cela ne concernait qu’une part assez faible des individus :



Environ 10% des clients seulement ne payent pas la totalité du prêt. Encore une fois, selon la finalité du projet et notre objectif, étant donné la faible proportion de contrats concernés, nous n’avons pas tenu compte de cela afin de ne pas complexifier la règle de décision initiale pour très peu voire aucun effet finalement.

2. Enrichissement : Vision dynamique.

Nous pouvons maintenant passer à l’échelle multi-mensuelle et définir un traitement permettant de déterminer si, finalement, le client retournera au paiement ou non ?

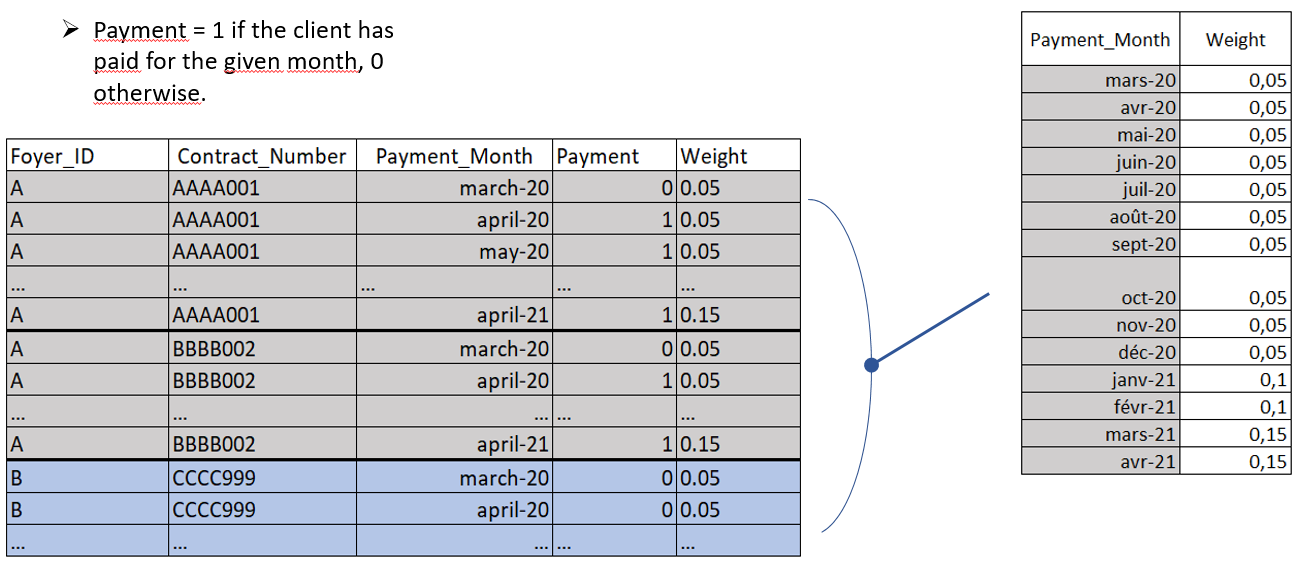
Nous parlions plus tôt de dynamique du paiement. Un client est susceptible d’avoir décidé de continuer à payer ou non les mensualités de son contrat depuis le début du moratoire, il peut également avoir essayé au départ avant de ne plus y parvenir financièrement.

*Pondération : Simili lissage exponentiel*

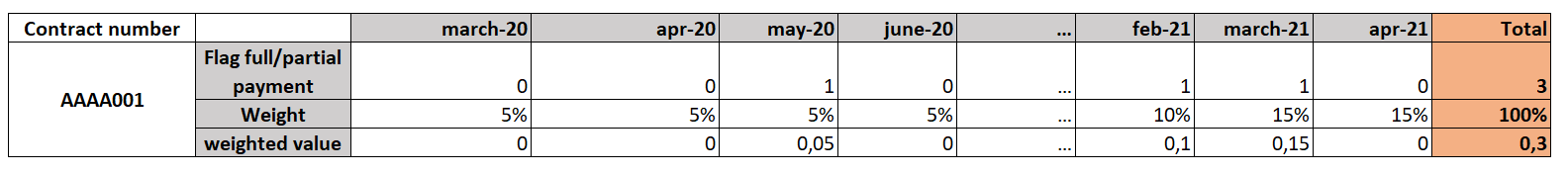
Nous avons donc décidé de calculer la variable cible étant le résultat d’une moyenne de paiement pondérée en fonction du mois considéré.

Le premier point était donc de déterminer les poids en question. Qu’est ce qui est le plus important pour nous afin de déterminer si un client va revenir au paiement ou non ? Le fait qu’il ait payé correctement quelques mensualités en début du moratoire avant d’arrêter, ou alors qu’il ait plutôt payé durant les derniers mois du moratoire quand bien même il avait des difficultés au début du moratoire ?

Nous avons opté pour cette seconde hypothèse. Dès lors, nous avons attribué un poids plus important au mois les plus récents afin de leur donner plus d’importance dans le calcul final de la target. Voici la forme que cela prend au final :



Voici maintenant comment cela se présente à l’échelle d’un seuil client :



La valeur finale de la moyenne pondérée est donc de 0,3.

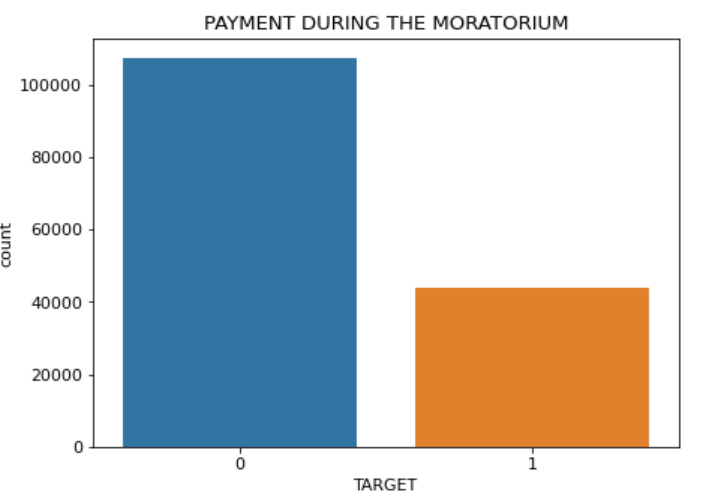
*Seuil de découpe : Bons/Mauvais clients*

Reste maintenant à déterminer avec les experts métiers de la Hongrie à quel niveau doit-on fixer le seuil de décision permettant de classer le client : 0,25 a été désigné comme tel.

Ainsi, si un client possède une moyenne pondérée supérieure à 0,25 il sera considéré comme susceptible à revenir au paiement sans que l’on n’ait besoin de l’appeler afin de l’y encourager.

Cette décision est le fruit de deux éléments différents :

* D’abord, la connaissance des indicateurs globaux de la Hongrie par les experts. A partir du moment où aucune justification d’impayé n’était demandée auprès des clients de Personal Finance en Hongrie, une sorte de catégorie de « payeur paresseux » a émergé comprenant donc des clients ayant arrêté de payer non pas pour des raisons financières, mais simplement par profit de la situation flexible à leur égard.
* De plus, avec ce seuil, nous sommes également parvenus à une variable dont la distribution, bien que déséquilibrée, s’avérait plus que raisonnable pour paramétrer un modèle efficace ayant suffisamment d’observations sur lesquelles s’entraîner :



C’est donc une répartition 70%/30% respectivement des « bons » / « mauvais » clients (les « mauvais » étant donc représentés par la classe 1, minoritaire, représentant 30% des observations) finalement.

II. Feature Engineering & Nettoyage de données

1. Mise au format client

*Justification méthodologique*

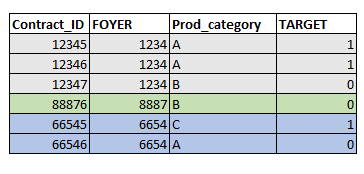
Comme déjà mentionné, l’objectif du modèle est de classer les individus en fonction de leur propension à revenir au paiement. Les contrats pour lesquels on suspecte qu’un encouragement au paiement sera nécessaire vont donc être appelés en priorité par les agents du recouvrement lors de la campagne d’appel.

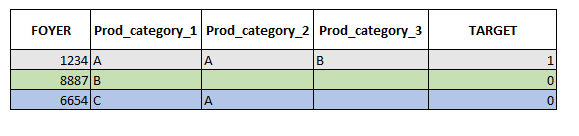
Cependant, il est évident que pour un client possédant plusieurs contrats (ces clients multi-contrat représentent 10% du portefeuille au total), on ne va pas engager des ressources pour le pousser à revenir au paiement du contrat A, en ignorant complètement le fait qu’il possède également un contrat B et C.

Nous avons donc décidé de passer la base en point de vue « foyer », on utilisera les termes « foyer » et « client » comme des substituts.

*Exemple d’application*

Voici un exemple de l’effet de cette conversion de la base :





S’agissant de la target, on adopte la même logique que pour la vision mensuelle en début de partie. Si sur la dynamique totale de Mars 2020 à Avril 2021 on considère qu’il a payé pour au moins 50% de ses contrats, on considère qu’il les a « globalement » tous payé.

La méthode de One-Hot encoding telle qu’appliquée ici permet donc de diminuer la dimensionnalité verticale de la base au détriment d’une verticalité horizontale. Mais l’effet ici est moindre étant le peu de variable concernée par cette transformation (4 variables au total).

Comme constaté ici, la variable Prod\_category\_1 n’a aucune valeur manquante, au contraire de Prod\_category\_2/3 qui ne concernent que les clients ayant respectivement, au minimum, 2 ou 3 contrats. On remplace alors les valeurs manquantes par un label « Missing ».

Une variable « Contract\_Count » est également créée afin d’énumérer le nombre de contrats possédés pour chaque client, cette variable va théoriquement venir absorber une partie de l’information située dans cette variable Prod\_category\_3 (Prod\_category\_2 étant sélectionnée dans notre modèle final).

En effet, ce n’est pas tant le produit possédé par le client dans Prod\_category\_3 qui compte, c’est plutôt le fait qu’il ait 3 contrats.

*Les différentes méthodes utilisées*

**SUM**

**One Hot Encoding**

**Max**

**Mean**

Pour les variables “Age”, “Revenus”, car potentiels différents titulaires de contrats différents au sein d’un même foyer

Pour les variables “mensualités”, “montant total dû”, prend compte de la totalité des contrats du client

Afin d’adopter le point de vue le plus récent, on maximise alors la notation du client ou son nombre d’enfants

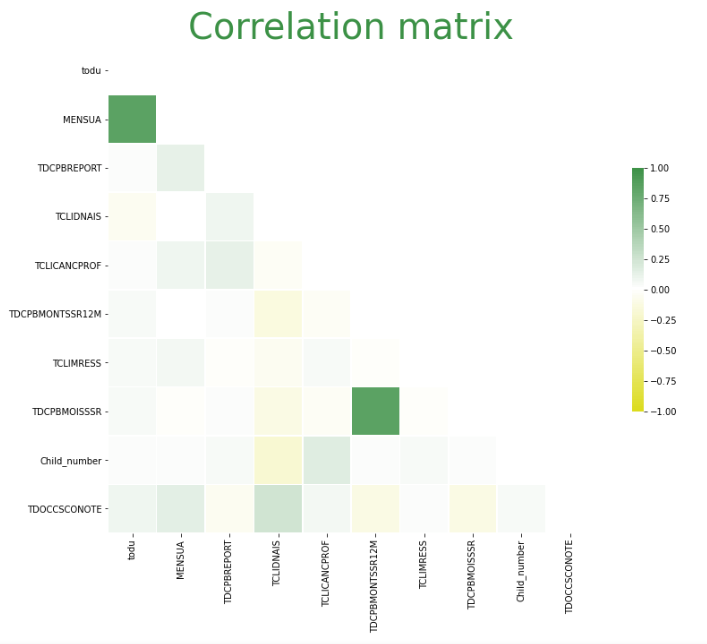
Pour les variables catégorielles

2. Nettoyage de données

Après la conversion de la base au point de vue « client ». Nous avons pu commencer l’étape de nettoyage de données (qui était en réalité quelque peu comprise dans la précédente partie).

*Etude des corrélations : Variables numériques*

La matrice de corrélation ci-dessous a permis d’isoler les couples de variables sur-correlées, le seuil de décision ici était de 0.6.



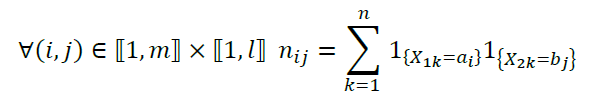
On remarque ici que deux couples de variables semblent dépasser le seuil de 0.6 que nous avons fixé.

Avant de décider d’en retirer chaque fois une des deux, une analyse métier est effectuée. En effet, ce n’est pas parce que le seuil est dépassé que la variable sera forcément supprimée. Nous allons d’abord vérifier qu’elles ont chacune un sens métier, parfois la corrélation peut être logique du fait qu’une variable dépend en réalité de l’autre. Si la relation est explicable, nous pouvons tout à fait garder les deux variables tant qu’elles ont un sens et que la métrique n’est pas excessivement proche de 1.

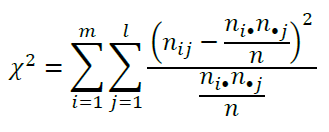
*Etude des corrélations : Variables catégorielles*

Le V de Cramer a également été utilisé afin d’étudier les corrélations des variables catégorielles. Le V de Cramer est donc une mesure de dépendance entre deux variables catégorielles. Sa valeur varie entre 0 (pour deux variables parfaitement décorrélées) et 1 (lorsque le lien est parfaitement linéaire entre les deux variables).

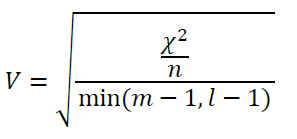
Prenons le cas de deux variables 𝑋1 et 𝑋2 respectivement avec des valeurs {𝑎1,𝑎2,…,𝑎𝑚} et {𝑏1,𝑏2,…,𝑏𝑙}. Pour un échantillons de n observations de ces deux variables, on dénote comme suit la fréquence observées de chaque pair de valeurs (𝑎𝑖,𝑏𝑗) :



La fréquence observes de 𝑎𝑖 est notée 𝑛𝑖, pour 𝑏𝑗 on note 𝑛•𝑗. La statistique du Chi-Squared est alors calculée comme telle :



Le V de cramer est alors obtenue avec la formule suivante :



Les seules variables étant ressorties de cette analyse s’agissant des variables issues du one-hot-encoding nous avons décidé de ne pas les retirer.

*Label encoding*

Nous avons décidé d’appliquer un traitement particulier aux variables catégorielles présentes dans la base, qu’elles aient été créées par nos soins ou non. En effet, nous avons décidé d’utiliser un modèle de Boosting en sortie pour traiter notre problématique.

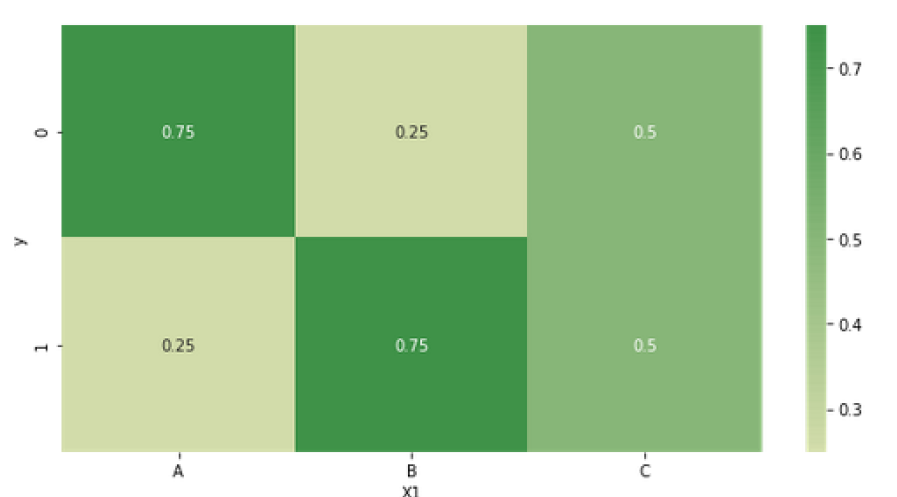
En l’occurrence, il s’agit du Catboost et celui-ci laisse la possibilité d’entrer en paramètres les variables catégorielles afin de leur appliquer un traitement particulier. Passer de variables qualitatives à des variables numériques grâce au label encoding a donc permis plusieurs choses en pratique au niveau de la sélection de variable :

* D’abord, d’utiliser l’algorithme de sélection de variables, que nous présenterons par la suite, qui est fondé sur l’algorithme des Random Forest qui nécessite des variables numériques.
* Il sera sans doute plus aisé pour la Random Forest d’apprendre avec un ordre logique qui ressort de ces variables catégorielles.

Finalement, cette méthode implique d’encoder les variables selon le niveau de taux de risque moyen associé à chacune des modalités de cette dernière.

Dans notre cas, le taux de risque est simplement la proportion de client qui sont susceptibles de ne pas revenir au paiement à la fin du moratoire, on va donc calculer cette métrique pour chacune des modalités d’une même variable catégorielles et les comparer.

Pour comprendre le processus, prenons une variable à trois modalités de la forme suivante :



On remarque donc qu’en termes de taux de risque moyen : B > C > A. Notre méthode va donc entraîner les transformations suivantes :

* B => 2
* C => 1
* A => 0

Le label encoding est donc une méthode qui peut rapidement être à l’origine de target leakage si on ne l’applique pas correctement, donc comme pour la totalité des méthodes de nettoyage de données qui implique la target d’une façon ou d’une autre : ces transformations doivent être appliquées sur l’échantillon d’entraînement, l’échantillon de test devant rester à l’écart de toute manipulation.

*Traitement des valeurs extrêmes*

Les valeurs aberrantes sont des observations pouvant biaiser les sorties des modèles car elles faussent les liens observés dans nos données. Elles doivent donc être traitées indépendamment du reste.

Elles peuvent être dues à une erreur de saisie ou à un effet exceptionnel propre à l’observation : En résumé, il s’agit d’évènements rares. Le traitement des valeurs extrêmes reste une étape importante et délicate car il faut traiter une information qui fausse nos observations mais qui reste une information malgré tout. Dès lors, se contenter de supprimer les valeurs extrêmes peut être assez contraignant.

Dans ce cas, j’ai traité les valeurs extrêmes en analysant la distribution de chacune des variables numériques. Ce à partir des quantiles à 5% et 95% auxquels je suis venu ajouter un intervalle de confiance de 1,5.

Dès lors, pour toutes les valeurs sortant de cet intervalle de confiance, propre à chacune des variables numériques on le rappelle, je remplaçais cette valeur aberrante par la valeur de la bonne supérieure ou inférieure afin de lisser la distribution de la variable et d’éviter de supprimer de l’information ou au contraire d’augmenter la dimensionnalité en créant une variable binaire afin de labeliser ces extremums.

III. Modélisation

1. Sélection de variable et boosting

*Sélection de variable*

Nous avons effectué notre sélection de variable grâce à la méthode RFECV (Recursive feature elimination cross-validated). Cette méthode indique quelles sont les variables les plus importantes grâce à une sorte de classement comme ce que peut l'on retrouver dans les arbres de décision.

Il s'agit, comme son nom l'indique, d'une méthode récursive qui, à chaque étape, va éliminer les variables les moins importantes (on a la main sur le nombre de variables que l'on veut retirer à chaque étape) ou les plus corrélées. Un modèle est construit à chaque étape et calcule la métrique de performance de notre choix (Nous avons opté pour le "f1 score " car notre variable cible a seulement 2 modalités, dont la classe minoritaire est notre cible principale), qui servira à la fin de comparatif pour choisir finalement le nombre de variables à garder.

Le "Cross-Validation" vient du fait que cette métrique de performance est, à chaque étape, calculée sur des plis de l'échantillon (on a aussi la main sur ce nombre de plis) et moyennisée, ce qui permet de s'assurer qu'on ne fait pas d’overfitting à certaines étapes du processus.

A la fin, nous avons choisi 15 variables comme étant le nombre optimal car aucun ensemble ne semblaient se distinguer en termes de performance.

*Modélisation*

L’objectif ici est avant tout de parvenir à estimer le mieux possible notre variable cible, l'interprétabilité du modèle n'est donc pas primordiale. Le recours aux modèles de stacking a été mis de côté étant donné leurs temps de calcul trop important et des résultats moyen relativement aux meilleurs algorithmes de boosting. Les réseaux de neurones quant à eux n'ont pas été sélectionnés car ils présentaient une performance moindre dans ce cas bien précis.

Ainsi, nous avons opté pour l'utilisation d'algorithmes de boosting car ces derniers sont généralement réputés pour leur grande performance face à des problèmes de ce genre. En effet, en regardant ce qui se faisait en matière de projet similaire, on retrouve régulièrement les algorithmes Xgboost, Light GBM ou encore les Catboost qui aboutissent à des performances excellentes relativement aux autres algorithmes.

Notre choix final s’est porté sur le Catboost pour différentes raisons :

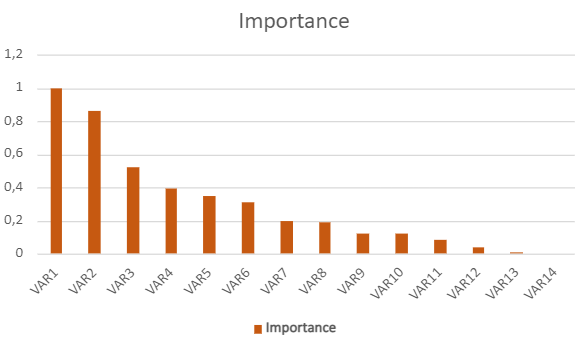
* Un traitement efficace des variables catégorielles qui constituaient la quasi-totalité des variables sélectionnées par notre algorithmes
* Un temps de calcul raisonnable, bien que plus important que les deux autres modèles cités
* De meilleures performances

*Importance des variables :*

On peut maintenant jeter un œil au classement des variables en fonction de leur importance dans la modélisation.

Comme souvent, ce tri est établi en fonction de l’importance de la variable. Lorsque l’on utilise le Catboost, la métrique permettant ceci est la LossFunctionChange.

* Pour l’obtenir, l’algorithme va simplement analyser la conséquence sur la Fonction de Perte qu’implique le fait de passer d’un scénario dit « normal » (où on inclue la variable dont on essaye de calculer l’importance) avec un scénario sans cette même variable (Ici on utilise le même modèle, mais on retire la variable de l’ensemble des arbres construits). Plus la différence sur la fonction de perte est élevée, plus la variable est considérée comme importante.



Afin d’obtenir ces valeurs de score d’importance, nous avons normalisé les résultats afin d’obtenir des valeurs imbriquées dans un intervalle [0 ; 1]. En l’occurrence, c’était surtout pour faciliter leur lecture par les collaborateurs opérationnelles. On perd les échelles de valeurs, mais pourtant on constate quand même assez aisément que le modèle est porté par 6 variables principales, les scores sont ensuite assez mineurs.

Pour autant, entraîner le modèle sur ces quelques variables qui semblent être déterminantes n’entraîne pas de meilleurs résultats sur les échantillons de test, au contraire. Nous avons donc privilégié cette version de la modélisation.

2. Interprétation des résultats : Point de vue opérationnel

Une phase importante du travail de modélisation dans ces secteurs réglementées porte sur l’interprétation des résultats.

Ici, ayant utilisé un modèle de type boite noire, je ne me suis pas pour autant essayé à l’utilisation des valeurs de Shapley afin d’éclaircir l’effet des différentes variables sur les résultats finaux car, encore une fois, l’interprétabilité des variables en tant que tel ne faisait pas partie des objectifs. J’ai ainsi privilégié l’interprétation des métriques finales à des fins opérationnelles.

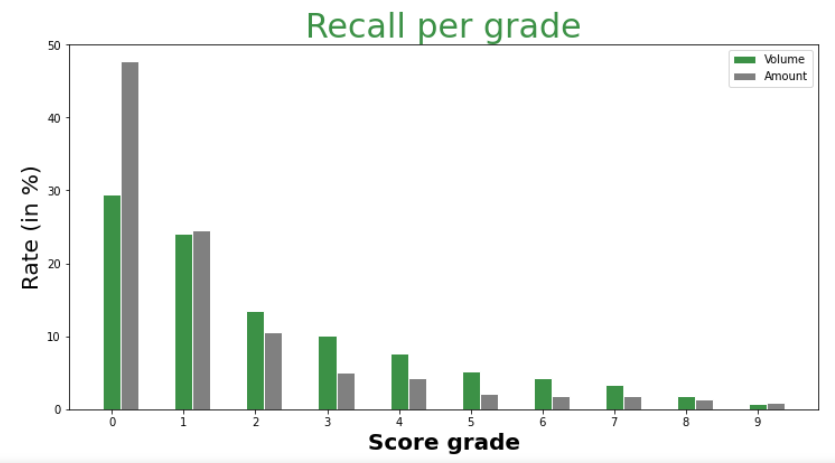
Lorsque nous obtenons les probabilités prédites d’appartenir à l’une ou l’autre des classes de la target, nous pouvons découper notre base de données en déciles afin d’obtenir une vision davantage granulaire de nos résultats, le découpage traditionnel à 0,5 (seuil de probabilité dans un problème binaire en-deçà/au-delà duquel l’observation est classée dans l’une ou l’autre des catégories) étant parfois quelque peu abrupt et n’aide pas à discriminer une population afin de prendre des décisions opérationnelles.

Ce qu’on peut dire c’est donc que l’on met complètement de côté l’aspect de classification (0/1) et nous ne nous concentrons que sur les probabilités brutes attribuées par le modèle.

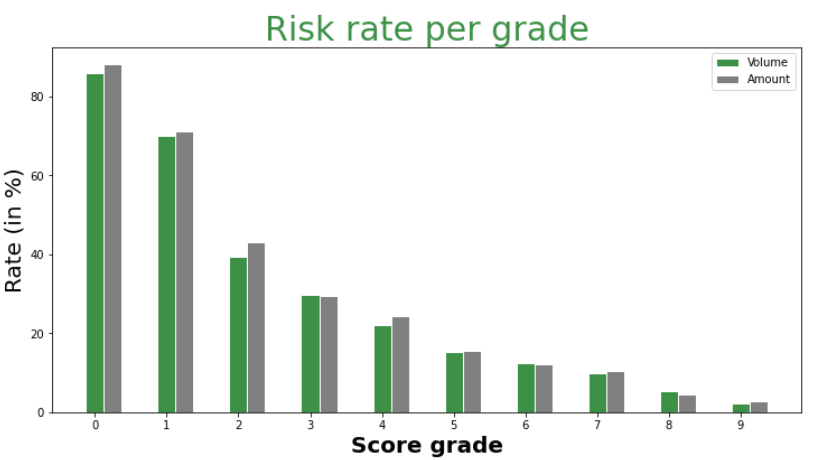
Face à une base de données déséquilibrée, les deux métriques principales que nous allons utiliser sont la Précision et le Recall. Nul besoin de préciser que ces métriques ont été obtenues sur l’échantillon de test.

*Précision et Recall : Une approche par décile*

Dès lors que les probabilités prédites sont coupées en dix déciles de tailles quasiment égales, il est simplement nécessaire de savoir que ci-dessous, le décile 0 est le décile comprenant les 10% des clients ayant la probabilité prédite d’appartenir à la classe 1, à savoir celle des individus qui sont le plus susceptible de ne pas reprendre le paiement à la fin du moratoire, la plus élevée. On parlera donc de décile le « plus à risque ».



* Le recall représente la capacité d’un modèle à capter correctement un % des observations d’une certaine classe. *Exemple : Pour 100 observations, avec une répartition 80/20 de fraudeurs/non fraudeurs. Si notre modèle prédit 30 individus comme fraudeurs, que les 20 « vrais » fraudeurs sont parmi ces prédictions, le recall sera de 100%, c’est la précision qui souffrira des quelques erreurs.*
* La colonne grise signifie que parmi 100% des montants impayés par ces clients, 48% d’entre eux sont captés par le décile le plus « risqué ».
* On monte même jusqu’à 80% de ces montants captés si l’on utilise les trois déciles les plus à risque.
* **La colonne verte quant à elle indique que parmi 100% des dossiers impayés, 30% d’entre eux sont dans le décile le plus « risqué ».** Un point positif ici est donc qu’on semble capter des individus qui représente davantage en termes de montant total dû qu’en effectif absolu. Les clients impliquant donc des montants empruntés importants sont donc efficacement captés par notre modèle.
* On atteint les 60% de ces dossiers si l’on utilise les trois déciles les plus à risque.



* Ici, la métrique analysée est la Précision. Celle-ci correspond à la capacité d’un score de ne pas se tromper lorsqu’il “prédit” qu’une observation appartient à une certaine classe. *Exemple (suite) : Dans le cas évoqué, notre précision serait de 20/30, car nous nous sommes trompés 10 fois dans notre prédiction de la classe minoritaire (les fraudeurs).*
* **Les colonnes vertes signifient que dans le décile le plus “risqué”, quand le modèle prédit que le client ne reviendra pas au paiement après le moratoire, il a raison dans 80% des cas.**
* Si l’on souhaite calculer cette métrique sur les trois premiers déciles, il est necessaire de procéder par la moyenne. On descend alors à environ 63% de Précision.

En combinant les résultats de notre modèle sur les trois premiers déciles, on constate donc que celui-ci est plutôt efficace et prometteur.

*Points d’accumulation*

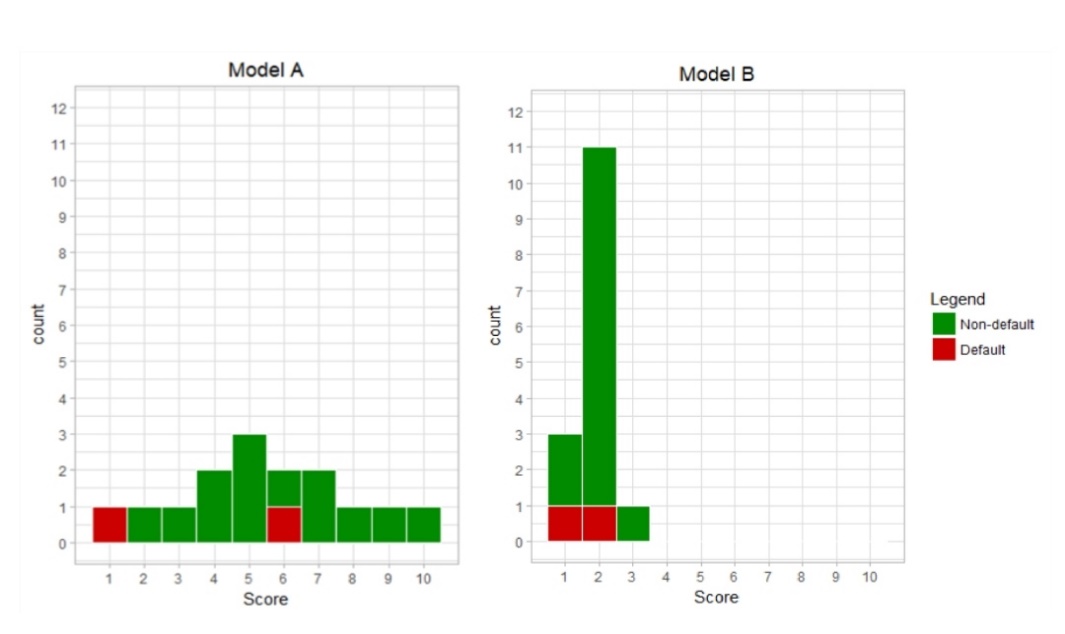
Ce type de graphe n’est pas vraiment relatif à une quelconque métrique de performance, il s’agit surtout d’un point de contrôle permettant de valider le modèle.

Les graphes de point d’accumulation sont nécessaires pour démontrer la sensibilité de notre score à différents niveaux de risque. Cela permet de démontrer son pouvoir discriminatoire et par extension sa capacité à classer le portefeuille de client en termes de score. C’est encore plus important quand l’on sait que la métrique de performance du Gini, particulièrement utilisée au sein des départements du Recouvrement et du Contentieux, a des difficultés à capter ces sensibilités. Cela signifie qu’il s’agisse de l’aspect discriminatoire et/ou de l’utilisation du Gini, l’analyse doit toujours être complété par ces graphes de point de d’accumulation.

Pour illustrer cela, prenons l’exemple du Scoring de Crédit, un score élevé signifiant dans ce cas que le client a profil de risque faible. A gauche nous avons le score en question, à droite la classe réelle d’appartenance :

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Model A** | | |  | **Model B** | | |
|  | **Score** | **Default** |  |  | **Score** | **Default** |
| 1 | 1 | 1 |  | 1 | 1 | 1 |
| 2 | 2 | 0 |  | 2 | 1 | 0 |
| 3 | 3 | 0 |  | 3 | 2 | 0 |
| 4 | 4 | 1 |  | 4 | 2 | 1 |
| 5 | 4 | 0 |  | 5 | 2 | 0 |
| 6 | 5 | 0 |  | 6 | 2 | 0 |
| 7 | 5 | 0 |  | 7 | 2 | 0 |
| 8 | 5 | 0 |  | 8 | 2 | 0 |
| 9 | 6 | 0 |  | 9 | 2 | 0 |
| 10 | 6 | 0 |  | 10 | 2 | 0 |
| 11 | 7 | 0 |  | 11 | 2 | 0 |
| 12 | 7 | 0 |  | 12 | 2 | 0 |
| 13 | 8 | 0 |  | 13 | 2 | 0 |
| 14 | 9 | 0 |  | 14 | 2 | 0 |
| 15 | 10 | 0 |  | 15 | 3 | 0 |

Si l’on calculait l’indice de Gini pour chacun de ces modèles, on obtiendrait 0,81. Cependant, on remarque aisément que le modèle A est beaucoup plus “sensible”. Regardons la distribution des scores :



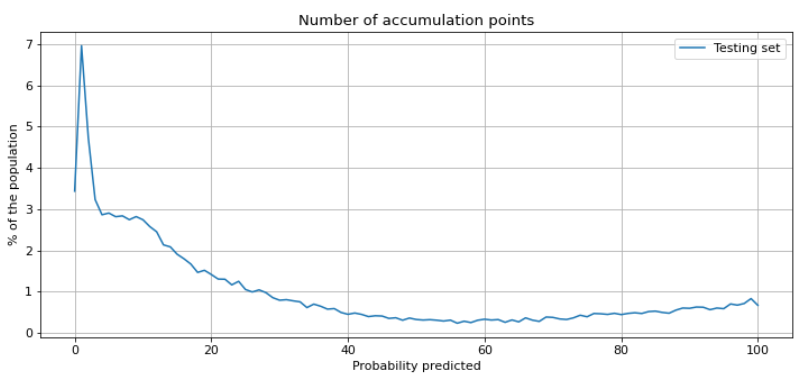
Imaginons maintenant le cas que chacun des Data Scientist derrière ces modèles décide de fixer son seuil de décision dans le but de capter 100% des clients à risque car l’on souhaite éviter à tout prix les faux négatifs (les clients qui seraient en réalité à risque mais non classés comme tel). Les matrices de confusion prendraient alors la forme suivante :

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Model A** | | **Actual** | |  | **Model B** | | **Actual** | |
| **Risky** | **Not Risky** |  | **Risky** | **Not Risky** |
| **Predicted** | **Risky** | 2 | 8 |  | **Predicted** | **Risky** | 2 | 12 |
| **Not Risky** | 0 | 5 |  | **Not Risky** | 0 | 1 |

Ce faisant, le modèle A raterait 8 dossiers non risqués contre 12 pour le modèle B. En termes de False Positive Rate, aussi appelé métrique de « spécificité », le modèle A serait à 62% contre 93% pour le modèle B.

Dès lors, c’est quelque chose dont on n’aurait pas pu se douter dans analyser le graphe d’accumulation et en se concentrant uniquement sur l’indice de Gini.

En entreprise, afin de s’assurer qu’il n’y aucune sur-représentation d’une quelconque valeur de score, on considère qu’aucune d’entre elles ne doit dépasser les 10% de la population. Jetons un œil aux résultats obtenus dans notre cas :



On constate alors que mis à part les notes de score très basses qui représentent un pic dans notre répartition, sans pour autant dépasser le plafond établi, les probabilités attribuées par le modèle sont bien réparties. Cela démontre plusieurs choses :

* Absence de phénomène d’entonnoir dans nos données, bonne capacité de discrimination de la population
* Absence d’une golden feature à l’origine de ce potentiel entonnoir, on s’assure ainsi une nouvelle fois (avec l’analyse des features importance) de se prémunir contre le phénomène d’overfitting
* Distinction d’une sorte de U dans la courbe, cela indique que le modèle ne donne que rarement des notes autour de 0.5, synonyme d’une forme d’indécision, même si comme indiqué en amont nous ne raisonnons pas vraiment autour de ce découpage traditionnel

**Conclusion**

Ce projet a donc été l’occasion pour la première de se plonger dans une problématique de modélisation dans l’entreprise et de prendre conscience de la réalité quotidienne de nombreux praticiens de la Data Science :

* Travail de nettoyage de données important
* Réflexion sur la construction d’une target qui n’existe pas en tant que tel au départ afin de répondre à une problématique
* Collaboration étroite avec des experts opérationnelles afin de déterminer les contours du sujet et ses guidelines, la communication est une clé importante du travail
* Nécessité de respecter la réglementation
* Prise en compte de l’environnement de mise en production

DAILY COLLECTION: REFONTE PYTHON (Centrale)

Cette dernière partie de mon rapport d’alternance prendra une orientation légèrement différente. Il ne s’agira plus ici d’expliquer et justifier un projet de modélisation dans son fond : pourquoi ai-je utiliser la standardisation plutôt que la normalisation pour mon data scaling ? Pourquoi la réglementation m’impose l’utilisation d’une régression logistique plutôt qu’un modèle black box ?

Cette partie me permettra surtout de retracer la création d’un outil de reporting hebdomadaire à destination des différents décisionnaires du service des Opérations : la genèse du projet, les objectifs, les conditions d’intuitivité et de facilité d’utilisation de l’outil, la paramétrisation, le format de présentation des résultats, la rapidité (optimisation de programmation) …

I. Programme Daily Collection Report

Un projet dénommé similaire avait déjà été mené par l’équipe Ops Analytics avant même mon arrivée au sein de l’entreprise. Il s’agissait d’un programme, au départ développé en VBA à destination des collaborateurs avec un profil « Métier/Business » et donc non initié à un quelconque langage de programmation.

L’équipe Ops Analytics avait alors créé un programme VBA à destination d’un profil utilisateur Métier (non technique). Ce programme permet de :

- A partir des données copiées quotidiennement par les différents pays du groupe Personal Finance sur un outil de gestion de base de données relationnelles qui effectue ensuite une série de contrôle, le programme va venir récupérer ces données grâce à Python

- Créer la base de données associée

- Générer le Reportings consolidé par pays et au global au format Excel (Stock, flux)

- Générer les graphes de suivi de plusieurs indicateurs (stock, flux)

L’objectif était donc de créer un outil capable de rendre compte quotidiennement de l’évolution des principaux KPIs que ce soit à travers une vision singulière pour chaque pays ou une vision consolidée à l’échelle du groupe.

Ce programme possédait cependant deux inconvénients principaux :

* Lenteur
* Maintenabilité difficile en raison du faible nombre d’individus utilisateurs de VBA dans nos services.

J’ai donc été chargé de participer à la refonte de ce projet sur le langage Python avec pour objectif de palier à ces deux inconvénients.

En plus de ces deux éléments, le programme se devait d’être fondé sur un haut niveau de paramétrage facilitant ainsi :

- L’ajout de nouvelles entités

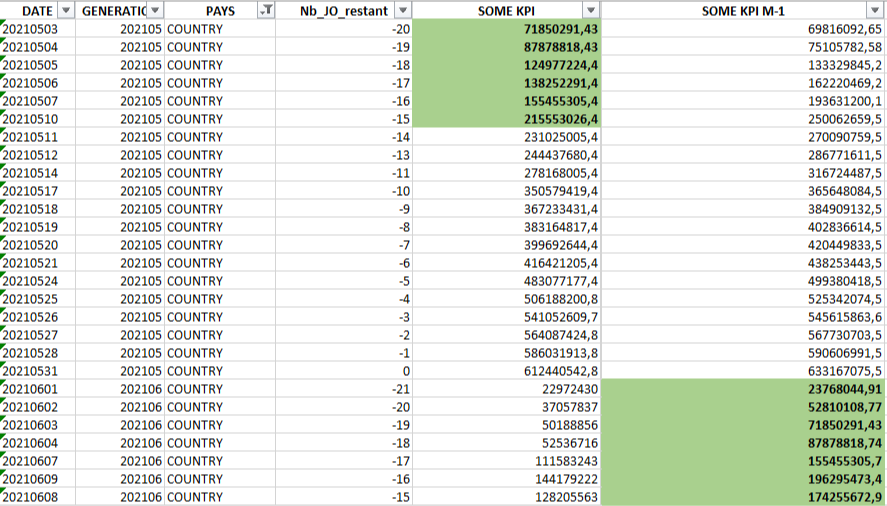
- L’ajout d’indicateurs dans le fichier Excel initial

1. Input et création de la base de données

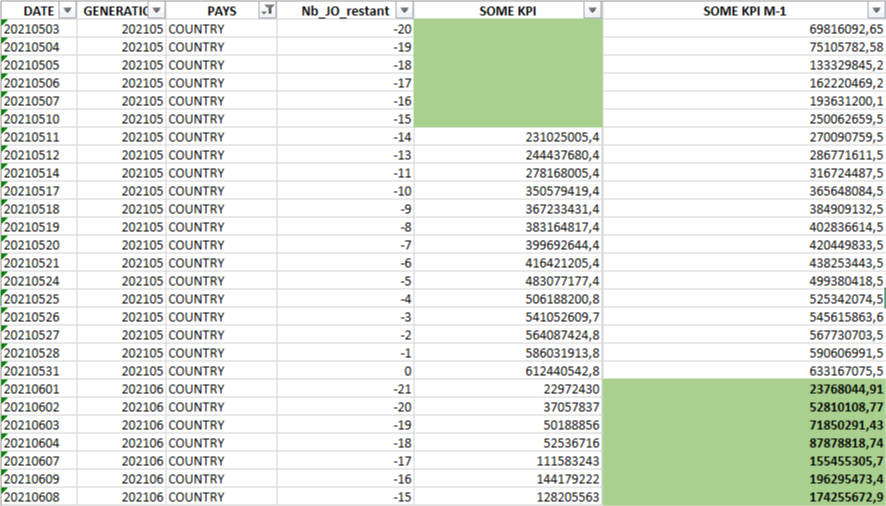
La base de données est générée chaque jour à partir d’une application de base de données relationnelle multiplateforme, à partir d’une liste de KPIs partagées par les pays quotidiennement.

Le programme ingère cette base de données et va procéder à un certain nombre de manipulations afin de rendre la base exploitable pour la suite :

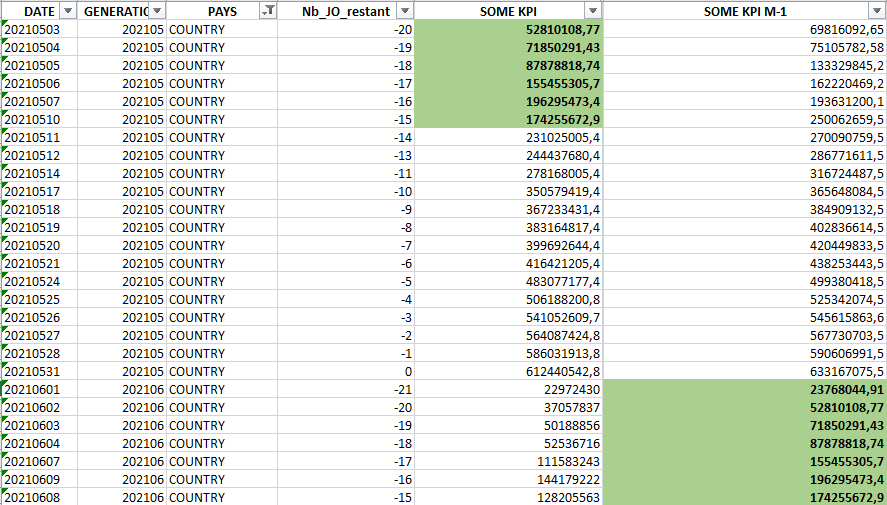
* Conversion en € des différents indicateurs de chaque pays en fonction d’un tableau mensuel de taux de change. Nous verrons plus tard que chaque run du script est fondée sur une date de travail choisie par l’utilisateur du script, cette date aura différents impacts sur le script, dont le mois choisi pour effectuer les conversions de la monnaie locale à l’euro.
* Actualisation des données tenant comptes des modifications que peuvent effectuer un pays d’un mois sur les données du mois précédent.



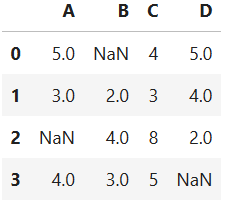
*Chaque pays, pour un mois M donné, donne les valeurs actualisées de certains KPIs en M-1 et M-2.*

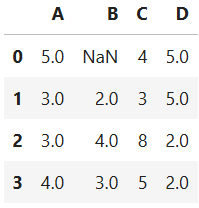


*Dans notre exemple, nous possédons les données actualisées de « SOME KPI » en pour les jours restants [-20 ; -15]. On va donc venir remplacer les anciennes valeurs dans le dataset.*



* Pointages des valeurs manquantes afin dans une base alternative pour alerte les utilisateurs de la récence des données qu’ils étudieront en sortie, ceci se fait également grâce à la colorisation de cellule possible grâce à Openpyxl. Dès lors, dans nos reportings, une cellule orange indique une valeur manquante qui a été remplie avec les données du jour précédent. Cette librairie a une place centrale dans ce projet, il s’agit d’un package Python permettant le traitement et la manipulation de données à l’échelle « cellulaire », cette librairie s’est avérée fondamentale pour mener à bien le projet en raison de la complexité du format des différents templates en sortie.
* Traitement des valeurs manquantes permettant d’assurer la soutenabilité du programme malgré l’absence parfois de données remplies par les pays. Cela prend simplement la forme suivante :





On n’effectue ce remplissage que sur le mois en cours. *Par exemple : Si le 10 Aout, un des pays ne nous remonte pas la valeur du KPI X, que la dernière fois qu’ils l’ont remonté c’était le 5 Aout, alors nous viendrons récupérer cette valeur du 5 Aout. Mais si la dernière fois qu’ils l’ont remonté était le 25 Juillet, là on ne récupère pas.*

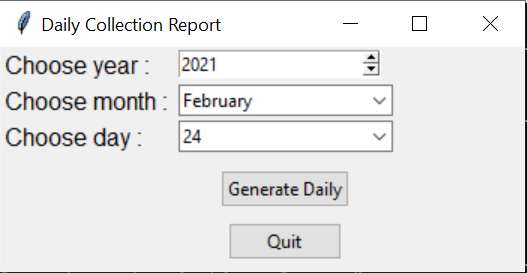
* La gestion des jours ouvrés et jours fériés

2. Fenêtre Tkinter & User experience

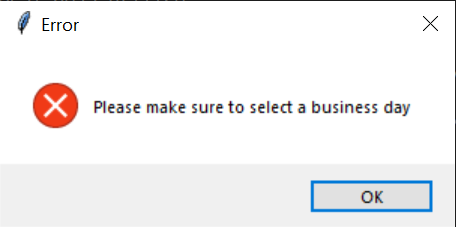
Le programme devait également s’avérer simple d’usage pour les potentiels utilisateurs :

* Choix des paramètres
* Création d’une interface graphique: Favorisation du clique-bouton pour éviter à l’utilisateur de rentrer dans le script ou un quelconque terminal de commande

*Choix de la date de travail avec Tkinter (Interface GUI)*

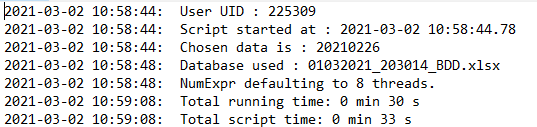


Le paramètre « Choose day » se plaçant automatiquement sur la date du dernier jour travaillé. Dans le cas où l’utilisateur choisit un jour non-travaillé (jour férié, weekend), le message suivant s’affiche et lui la possibilité de choisir une nouvelle date :



*Journal de bord*

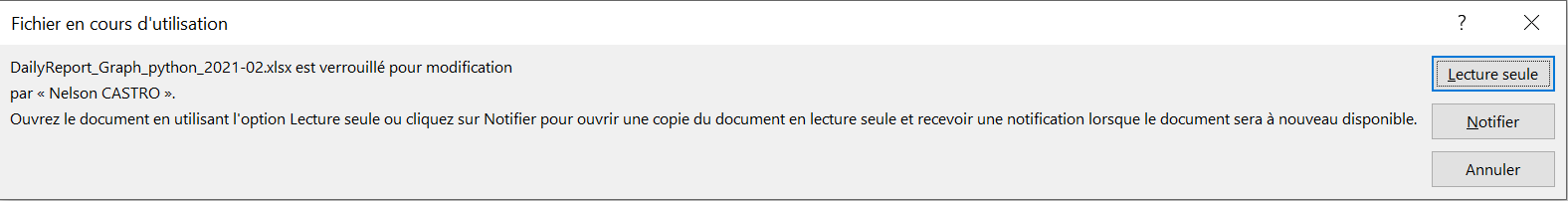
* Transparence des procédures de contrôles en background.
  + Utilisation de fichier .log afin de rendre compte de chaque utilisateur du programme. Ces fichiers prennent la forme suivante :



Ainsi, on sait grâce à l’ID de l’utilisateur qui a lancé le script, à quelle date/heure et pour quel jour, la base qui a été sélectionnée pour effectuer les calculs ainsi que le temps que le script a mis à « tourner ».

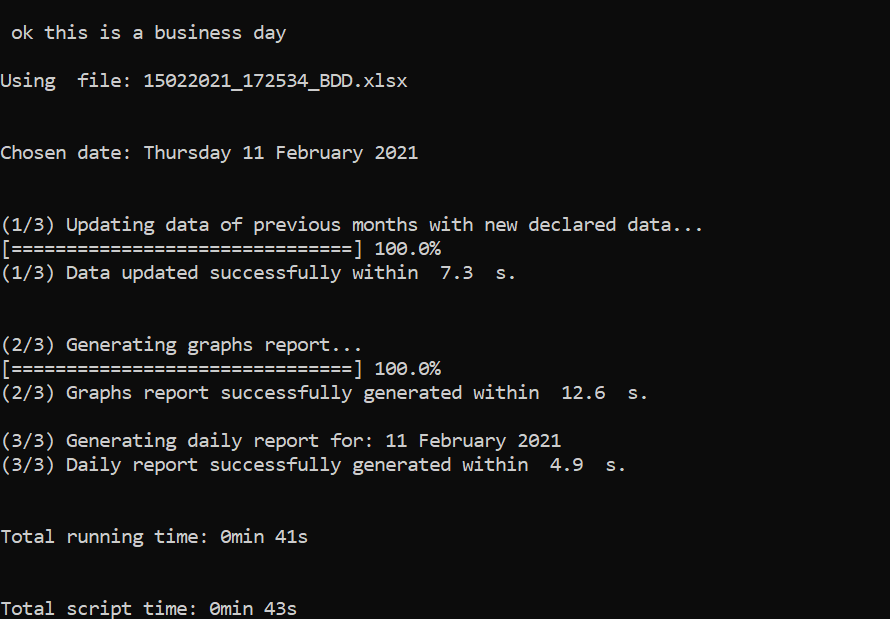
*Travail dans un environnement partagé*

* + Les données en input se trouvant sur un disque réseau partagé ont pu poser un problème lors de la manipulation de fichiers Excel déjà ouverts par un autre utilisateur :
    - Un contrôle est donc effectué et permet de remonter l’utilisateur en question grâce à un usage de la commande os.rename(name\_of\_file, new\_name\_of\_file) permettant, comme son nom semble l’indique, de renommer un fichier directement depuis Python et qui, surtout, entraîne une erreur si le fichier à renommer est déjà en ouvert. Si cette condition est validée, le programme ouvre tout de même le fichier concerné de force, ce qui amène ce message :



* + D’autres contrôles sont effectués pour vérifier la présence des pays d’intérêts, des KPIs utilisées, le calcul des jours travaillées restants à la date t.

* Une fois que toutes les conditions requises ont été remplies, le programme va manipuler les données et générer les Reportings en Excel en output. Un exemple de ce qui apparaît sur l’écran de l’utilisateur durant le processus :



On voit donc que le script est divisé en trois étapes majeures, les barres de chargement se remplissent en temps réel afin d’indiquer à l’utilisateur l’avancée des étapes.

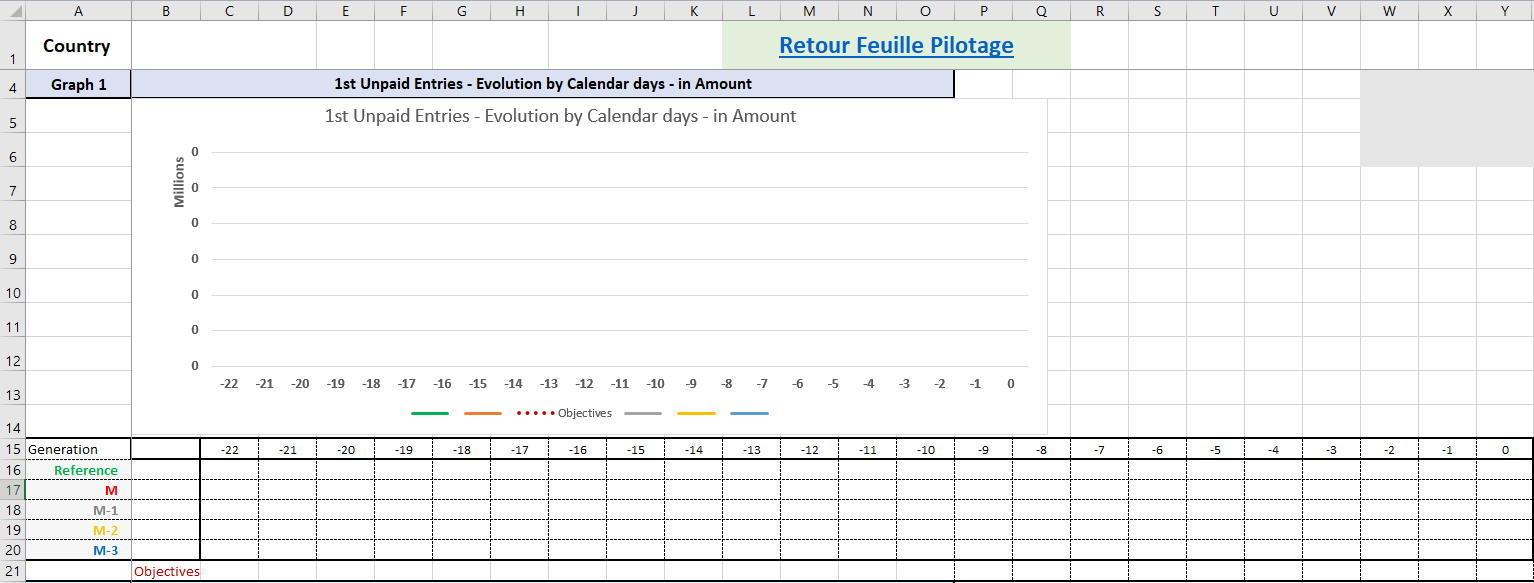
II. Génération de reportings et graphes

*Création et organisation des templates*

Suite à la création de la base de données, le programme va créer les Reportings en Excel à l’aide de OpenPyxl.

Aujourd’hui, il existe 3 Reportings majeurs

* Une photo quotidienne par pays et en cumul (nombre et montant) de l’ensemble des indicateurs remontées.
  + Cette photo effectue des calculs intermédiaires et créé à la volée des indicateurs (Notamment des efficacités, ROI et autres indicateurs de pertes).
  + Si par exemple un pays ne remonte pas les données à J (retard ou jour chômé ou weekend), le programme récupère les données de la veille pour la vision stock et initialisé à zéro les données flux.
* Suivi du flux quotidien des indicateurs principaux à l’échelle Pays avec une vision depuis le début du mois.
  + Graphes en jour ouvré de plusieurs indicateurs avec comparaison M-1, M-2, M-3 et M de référence : par pays et au global



Voici un exemple du template créé dans le fichier des graphes par pays. Chaque tableau est connecté au graphe linéaire au-dessus. Le tableau est alors rempli case par case grâce à OpenPyxl, ce qui va remplir le graphe. Plusieurs couples Tableau + Graphes sont ainsi répartis sur chaque onglet du fichier (Un onglet = Un pays), c’est ici que la librairie utilisée est importante, afin de faciliter la manipulation du document cellule par cellule.

Pour faciliter le travail en amont, ces tableaux sont mis au format grâce au code Python. Les colonnes représentent les jours travaillés restants (avant la fin du mois), l’index lui représente le mois d’étude (M), le mois de référence (M-12) et les trois mois précédents. Cela permet ainsi d’avoir une vision de la tendance des différents KPIs.

Cette approche en termes de jours de travail restants permettant d’éviter au maximum les décalages de calendrier d’un mois sur l’autre, mais également les jours fériés qui peuvent avoir lieu dans certains pays seulement.

III. Environnement technique du programme.

Le programme permet, comme indiqué, d’être lancé en un clic, par des utilisateurs opérationnels qui n’ont ainsi pas à rentrer dans le script et voir une seule ligne de code.

*Organisation du script*

|--- logs

| +-- Contains all “DD\_MM\_YYYY\_HH\_MM\_SS.log” files, description later.

|--- objectifs

| +-- **DailyReport\_Objectifs.xlsx**

|--- old

| +-- Old version of the script kept for safety purpose.

|--- parametrage

| +-- **DailyReport\_Graph-V6-modele-tous-pays.xlsx**

| +-- **daily\_python\_param.xlsx**

| +-- **DailyReport\_devise.xlsx**

|--- programme

| +-- **DailyCollectionReports.bat**

|--- recently\_generated

| +-- Daily\_report\_python\_all\_days.xlsx

| +-- **Daily\_report\_python\_YYYY-MM.xlsx**

| +-- **DailyReport\_Graph\_python\_YYYY-MM.xlsx**

|--- script

| +-- **Interface.py**

| +-- src (folder)

| +-- **Main.py**

| +-- countryfunctions.py

| +-- dailyfunctions.py

| +-- fonctionsmef.py

| +-- librairies.py

| +-- paths.py

| +-- Variables.py

|--- README.md

*Utilisation*

Comme mentionné en amont, ce script a été conçu afin de faciliter le mieux possible l’expérience utilisateur. Quand bien même nous usons de méthodes de programmation modulaire afin de diviser le script main.py en plusieurs fonctions et packages qui exécutent différents éléments du programme, et que les templates Excel de référence sont également nombreux, jamais un utilisateur n’aura à se préoccuper d’autre chose que du fichier suivant : **DailyCollectionReports.bat**

En cliquant dessus, ce batch lance l’interface GUI présente dans le script interface.py qui va elle-même appeler le script main.py et les différents packages associés, les contrôles sont ainsi effectués en arrière-plan et l’utilisateur n’a qu’une chose à faire : Choisir la date de travail. Le programme s’occupe ensuite d’aller récupérer les données en Input, les plus récentes à chaque fois, et exécute le reste du travail.

**Conclusion**

Ce projet m’a donc permis de prendre pendant quelques semaines mes distances avec les missions de modélisation.

* La construction de reporting s’est avéré être un exercice très captivant : j’ai pu collaborer de façon très régulière avec des experts opérationnels, j’ai ainsi pu me mettre dans leur position afin d’essayer de comprendre leurs besoins. J’ai donc pu exercer un rôle de support dans lequel tout était tourné vers l’utilisateur final, un être humain, contrairement à la majorité des modèles que nous pouvons construire et qui sont à destinateur de notre outil, donc une machine, d’implémentation et d’intégration des scores.
* Le projet a également été un défi technique du point de vue des techniques de programmation utilisée : nous sommes passé d’un script VBA tournant en un peu plus de 20 minutes à un script Python qui tourne en 45 secondes.

CONCLUSION GLOBALE

Ce fut donc une année riche en expériences diverses. J’ai pu toucher à un panel élargi des missions traditionnels du Data Scientist : Modélisation, Analyse descriptive et Reporting.

J’ai également pu évoluer dans un environnement très enrichissant au sein des fonctions centrales en travaillant régulièrement avec des collaborateurs au sein des filiales du groupe BNP Paribas Personal Finance : en particulier Brésil, Espagne et Chine.

Dans une équipe bienveillante et spécialiste des problématiques des opérations : J’ai pu bénéficier de leur expérience et progresser sur le fond tandis que ma formation m’apportant les outils nécessaires sur le plan technique.

Ces différents éléments ont pu créer un environnement épanouissant au sein duquel j’ai pu faire mes premiers pas dans la pratique de la “Data Science” en entreprise.

Annexe

