Mémoire présenté le :

pour l'obtention du Diplôme Universitaire d'actuariat de l'ISFA
et l'admission à l'Institut des Actuaires

Par : Valentin Chanteloup

Titre : Approche Machine Learning pour la prédiction des rachats en épargne

Confidentialité : ☐ NON ☒ OUI (Durée : ☐ 1 an ☒ 2 ans)

Les signataires s'engagent à respecter la confidentialité indiquée ci-dessus

Membres présents du jury de l'IA *Signature*

.....

Entreprise : CNP Assurances
Nom :
Signature :

Membres présents du jury de l'ISFA

Directeur de mémoire en entreprise :
Nom : Mme. Ndeye Marie GUISSÉ
Signature :

Autorisation de publication et de mise en ligne sur un site de diffusion de documents actuariels *(après expiration de l'éventuel délai de confidentialité)*

Signature du responsable entreprise

Signature du candidat

Sommaire

1. INTRODUCTION 1

1.1. Généralités 2
1.1.1. L'assurance vie 2
1.1.2. Les supports d'investissement en assurance vie 3
1.1.3. La fiscalité de l'assurance vie 4
1.1.4. Le risque de rachat 7

1.2. Approche retenue pour la modélisation du rachat 8

2. CONSTRUCTION DE LA BASE DE DONNEES ET ANALYSE DESCRIPTIVE 11

2.1. Présentation des données utilisées 12
2.1.1. Extraction et retraitement des données 13
2.1.2. Agrégation des données 18
2.1.3. Création des features 24

2.2. Statistiques descriptives 26

2.3. Descriptif du jeu de données final 36

3. APPROCHE N°1 : MODELISATION PAR REGRESSION 40

3.1. Introduction au Machine Learning 41
3.1.1. Cadre théorique 42

3.2. Approche de notre modélisation 46
3.2.1. Echantillonnage 46
3.2.2. Choix des hyperparamètres 48
3.2.3. Premier modèle : Utilisation de la Root Mean Squared Error 51
3.2.4. Second modèle : MSE avec fonction de perte personalisée 58
3.2.5. Troisième modèle : Modélisation avec fonction de surive 65

4. APPROCHE N°2 : MODELISATION PAR CLASSIFICATION 78

4.1.Cadre de l'approche 79
4.1.1. Calcul d'un score d'appétence au rachat 79
4.1.2. Echantillonnage 80

4.2. Construction des nouvelles bases 81
4.2.1. Architecture de la nouvelle base 81
4.2.2. Définition de la cible (*Target*) 82
4.2.3.Récupération et agrégation des nouvelles variables 82

4.3. Modélisation et résultats 88
4.3.1. Modèle utilisé 88
4.3.2. Métriques 89
4.3.3. Résultats 90
4.3.3. Interprétation 95

4. CONCLUSION 97

5. BIBLIOGRAPHIE 100

Résumé

Mots clés : Assurance vie, produit épargne, risque de rachat, modélisation de durée, censure, machine learning, arbre de régression, algorithme XGBoost, fonction de perte personnalisée, modélisation de survie, arbre de classification, scoring individuel.

En assurance épargne, l'option de rachat permet à l'assuré de retirer en partie ou en totalité le capital investi sur son contrat à tout moment, cela constitue un risque important pour l'assureur.

Le risque de rachat est très complexe de par sa nature conjoncturelle et structurelle et il n'est toujours évident à modéliser par une approche actuarielle classique. Cette situation se complexifie également par l'évolution de la fiscalité qui peut impacter les comportements des assurés en matière de rachats. En raison de ces contraintes, il peut être pertinent d'utiliser le maximum d'informations disponibles afin de pouvoir exploiter au mieux les signaux sur les intentions de l'assuré.

En ce sens, une des interrogations en amont de ce travail s'est portée sur l'apport d'une approche par *Machine Learning* à l'estimation du risque de rachat par rapport à la modélisation classique des lois. L'idée retenue dans le cadre de ce mémoire est de proposer différentes approches pour prédire le moment où l'assuré serait le plus susceptible de racheter son contrat.

La première approche consiste à développer un modèle pouvant prédire une durée jusqu'au rachat avec les informations disponibles à la souscription du contrat, grâce à l'appui d'algorithmes de *Gradient Boosting* dont certains pouvant prendre en compte les données censurées.

Dans un second temps, l'étude a été élargie pour pouvoir prendre en compte les données de la vie du contrat. Une approche dans laquelle le but escompté est l'identification de manière individuelle du risque de rachat à court terme pour chaque contrat d'assurance-vie.

Il en ressort que les données disponibles à la souscription du contrat ne permettent pas à elles seules de fournir un signal suffisant afin de modéliser individuellement le risque de rachat, la seconde approche permet d'obtenir de meilleurs résultats.

Abstract

Key words: Life insurance, saving life-insurance product, lapse risk, survival time modelling, censoring, machine learning, regression tree, XGBoost algorithm, custom loss-function, survival analysis, classification tree, individual scoring.

In saving insurance, the possibility of surrender allows the policyholder to withdraw a part or all of the capital invested in the policy at any given time, and thus constitutes a significant risk for the insurer.

Lapse risk is very complex due to its nature and is not always easy to model it using a traditional actuarial approach. This situation is also getting more complex with the evolution of taxation, which may impact policyholders' surrender behaviour. In such context, it is relevant to use the maximum amount of data available in order to be able to better exploit the signals on the policyholder's intentions.

To serve this purpose, one of the questions that arose in the early stages of this work concerned the contribution of a Machine Learning approach to the estimation of the lapse risk in relation to the classic law modelling. The idea behind this study is to offer different approaches to predict the moment when the insured is the most likely to surrender his policy.

The first approach consists in developing a model that can predict a duration until the policy surrender, with the information available at contract subscription time thanks to the support of Gradient Boosting algorithms, some of which can take into account censored data.

In a second step, the study was extended to consider data from the life of the contract. An approach in which the expected goal is the individual identification of the short-term lapse risk for each policy.

Finally, the study shows that the data available at the time the contract was taken out is not sufficient on its own to provide a sufficient signal to model lapse risk individually, the second approach provides better results.

Remerciements

Je tiens d'abord à remercier ma tutrice, Ndeye Marie GUISSÉ, pour la confiance qu'elle m'a accordé durant cette année mais également pour sa patience, son soutien et pour tous ses conseils lors de la rédaction de ce mémoire.

Je remercie également toute l'équipe du DataLab de CNP Assurances : Romain MERIDOUX, J.C., Jacques Arthur MOMBO, Gabriel ROGOSIC, Hervé TRINH, Benoit PIVETEAU, David ZON, Annie TOGLOZIN, Zacharie BENNINI et Marko MACANOVIC, pour les échanges que nous avons pu avoir au cours de cette année, professionnels ou non, mais toujours agréables. Aussi je souhaiterai exprimer ma reconnaissance à Ievgen SAVIN et à Baptiste DIELTIENS qui ont pris le temps de se rendre disponible quand cela était nécessaire et ont pu par leurs conseils avisés répondre à toutes mes interrogations.

Je tiens également à remercier tous les professeurs, les intervenants et toute l'équipe pédagogique et administrative de l'ISFA pour cette année passée dans un contexte unique mais riche.

Enfin, mes remerciements sont adressés à toutes les personnes qui m'ont aidé lors de la réalisation de ce mémoire, en particulier ma famille pour leur grand soutien et Quentin LOPEZ pour les échanges que nous avons pu avoir sur nos sujets respectifs.

Chapitre 1

Introduction

1.1. Généralités

1.1.1. L'assurance vie

Un contrat d'assurance vie est un contrat dont les engagements pris par l'assureur et le souscripteur sont à durée de vie humaine. Le marché de l'assurance vie est principalement composé des contrats d'épargne, retraite, et prévoyance. Les contrats restants représentent une faible proportion des encours.

En d'autres termes, un contrat d'assurance vie est un contrat par lequel l'assureur s'engage, contre le paiement de primes par l'assuré ou le souscripteur, à verser un capital ou une rente à une ou plusieurs personnes déterminées. On peut distinguer 3 éléments déclencheur de la garantie :

- Contrat « en cas de vie » : le contrat prévoit ici le versement du capital constitué ou de la rente si l'assuré est toujours en vie à terme du contrat. Ce type de contrat sert un objectif d'optimisation de l'épargne en termes de fiscalité à long terme.

- Contrat « en cas de décès » : le souscripteur constitue une épargne au profit d'une tierce personne. Le contrat précise qu'au décès du souscripteur, un montant de capital sera versé au bénéficiaire de son choix. Ce type de contrat de prévoyance sert un objectif de préparation de succession.

- Contrat « mixte » : à l'échéance du contrat, le versement d'un capital ou d'une rente est garanti, soit au souscripteur s'il est en vie, soit à un bénéficiaire si le souscripteur est décédé.

Un contrat d'assurance vie peut être investi sur différents supports d'actifs selon les besoins de l'assuré.

1.1.2. Les supports d'investissement en assurance vie

Pour les contrats d'assurance vie, il existe principalement deux types de supports pouvant répondre à différents besoins du souscripteur.

Tout d'abord, il y a le support en euros, l'assuré bénéficie dans ce cas d'un capital garanti faisant l'objet d'une revalorisation permanente, sa valeur ne pouvant pas baisser. Néanmoins, les performances de ce type de support sont limitées (+1.1% en 2021 alors que la moyenne de l'assurance vie était de +3.1%). Il est destiné aux personnes les plus averses au risque.
Le deuxième est le support en unités de comptes, où l'assuré s'engage sur le nombre de parts et non sur sa valeur. L'épargne ainsi investie dans ce type de support n'est pas garantie car sujette aux fluctuations des actifs. Néanmoins ce type de support est depuis plusieurs années plus performant. En 2021, 39% des cotisations ont été faites sur ce type de support.

Dans le cadre d'un support en euros, c'est l'assureur qui porte tous les risques puisqu'il s'est engagé à fournir un rendement défini. Ce rendement est obtenu par le réinvestissement des fonds placés sur le support. Cependant pour un contrat en unités de comptes c'est l'assuré qui porte le risque, l'assureur s'octroyant une partie de la plus-value de l'actif.

On dit d'un contrat qu'il est mono-support lorsque le capital investi porte sur un unique actif, il s'agit généralement de supports en euros. A l'inverse, les contrats multisupports sont des contrats où les capitaux sont placés sur divers supports, permettant ainsi à l'assuré de diversifier son épargne. La majorité des contrats multi-supports combinent des supports en euros et des supports en unités de compte, ce qui permet de prendre une approche prudente face au risque sans pour autant perdre en performance.

Il est également important de noter qu'en France, toute plus-value dégagée par un contrat d'assurance-vie est soumise à une imposition. Nous détaillerons ce point dans la section suivante.

1.1.3. La fiscalité de l'assurance vie

On peut définir le produit d'un contrat d'assurance vie comme étant la différence entre la somme des remboursements effectués au souscripteur et la somme des versements effectués par celui-ci. Pour tout événement de sortie de capital, les produits financiers dégagés sont soumis à une imposition. En France, cette imposition s'effectue au titre de l'impôt sur le revenu.

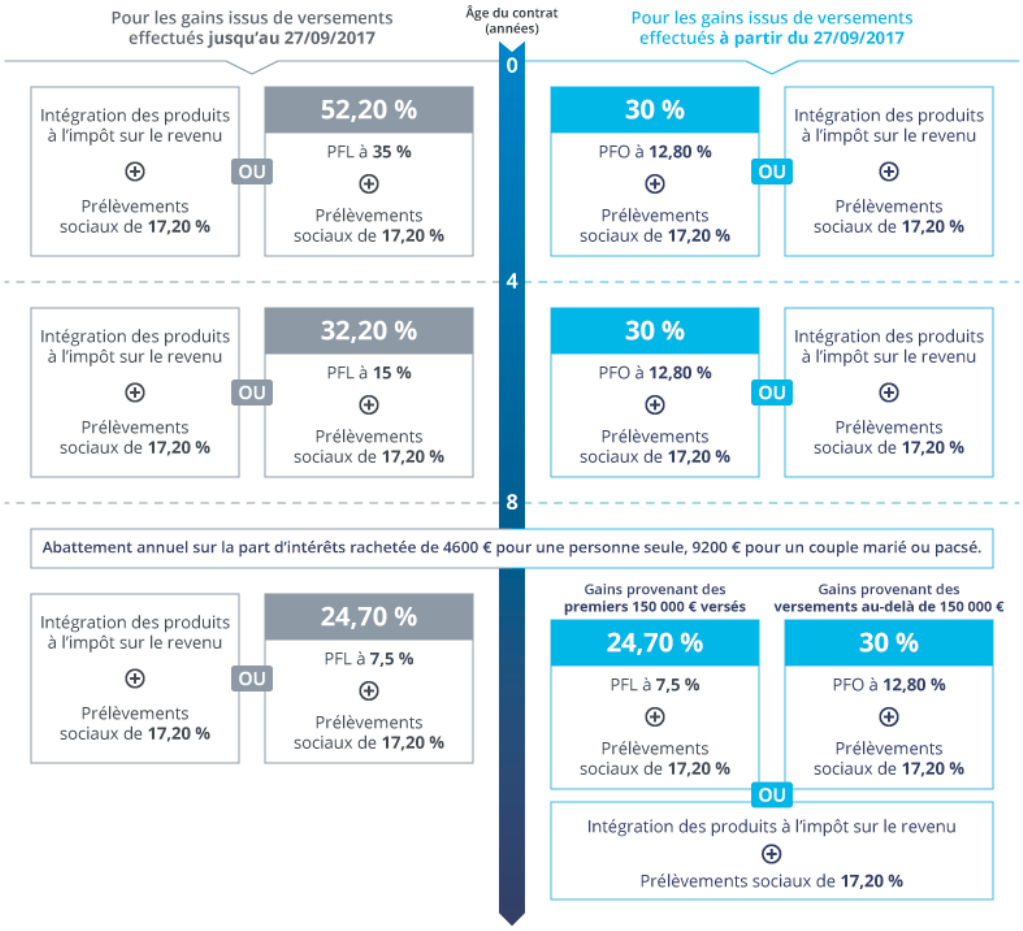

Illustration 1 : Fiscalité s'appliquant aux produits épargne

(source : *https://placement.meilleurtaux.com/*)

Comme on peut le constater avec le schéma précédent, avant l'implémentation de la nouvelle fiscalité communément appelée la « *flat-tax* » ou « Prélèvement Forfaitaire Unique » (PFU), le taux d'imposition suivait un barème dégressif selon l'ancienneté du contrat. Ainsi, la fiscalité est

plus avantageuse à partir de la 4ème année et elle l'est plus encore à partir de la 8ème. De ce fait, la fiscalité incite le souscripteur à conserver son épargne jusqu'à la huitième année. C'est un point sur lequel nous reviendrons par la suite.

Depuis le 27/09/2017, la « *flat-tax* » est entrée en vigueur et modifie la fiscalité des contrats d'épargne, son but est de simplifier et d'alléger la fiscalité. Elle consiste à imposer tous les revenus du capital à un taux forfaitaire de 30%. Cela concerne tous les gains issus des versements effectués à partir de la date d'entrée en vigueur de la nouvelle fiscalité.

Pour certains contrats multi-supports, l'évolution de la législation a permis la création de certains mécanismes d'incitations fiscales, on peut citer entre autres :

- Les contrats PEP : Les contrats PEP (où Plan d'Epargne Populaire) sont des produits d'épargne créés dans les années 1990 pour inciter les Français à épargner sur le long terme. Ainsi, fiscalement, ces contrats bénéficient d'une exonération de l'impôt sur le revenu à partir de la 8ème année de vie du contrat. Ces produits ne sont plus commercialisés depuis le 25 septembre 2003, mais toute personne ayant souscrit avant cette date peut continuer de profiter de ses avantages. Il est également possible de transférer son contrat PEP d'un établissement financier à un autre tout en conservant son antériorité fiscale.

- Les contrats DSK : Créés en 1998, les contrats DSK sont des contrats en unité de compte dont 50% des encours au minimum sont investis dans des actions françaises et européennes, et qu'au moins une part de 5% est investie dans des actifs dits risqués (titres de sociétés non cotées, etc..). Il n'est plus possible de souscrire à ce type de contrat depuis le 1er janvier 2005 bien que ces contrats puissent toujours être alimentés par leurs souscripteurs. D'un point de vue fiscal, et à partir de la 8ème année de détention, le titulaire du contrat bénéficie d'une exonération totale de l'impôt sur les revenus de ces contrats.

- Les contrats de capitalisation : Il s'agit de produits d'épargne mono-support ou multi-supports. Avant janvier 2018, date de remplacement de l'impôt sur la fortune (ISF) par l'impôt sur la fortune immobilière (IFI), les contrats de capitalisation permettaient la non-imposition des plus-values réalisées au titre de l'ISF. Depuis cette date, l'avantage du contrat de capitalisation est qu'il n'est pas soumis à l'IFI, autrement ces contrats sont soumis aux mêmes prélèvements et impositions que les autres contrats d'assurance vie.

Il existe néanmoins d'autres mécanismes d'incitation fiscale à destination des détenteurs de contrats d'assurance vie pouvant leur permettre de modifier leurs contrats tout en conservant leur antériorité fiscale, il en existe plusieurs types :

- Les transferts FOURGOUS : L'amendement Fourgous instauré en juillet 2005 a pour but de réorienter les contrats en support euro vers des contrats en unité de compte ou multisupports, disposant ainsi d'un rendement plus élevé. Il donne la possibilité à l'assuré de transformer son contrat mono-support en un contrat multi-supports sans perte de l'antériorité fiscale, en transférant *a minima* 20% de l'épargne dans des supports en unités de compte.

- Les transferts PACTE : La loi Pacte du 22 Mai 2019 permet aux titulaires de contrats épargne de transférer leurs avoirs d'un contrat à un autre si celui-ci est commercialisé par la même compagnie d'assurance, tout en conservant leur antériorité fiscale. Ainsi, l'assuré peut choisir de transférer ses fonds vers un contrat plus performant, tout en conservant les avantages fiscaux de son ancien contrat.

1.1.4. Le risque de rachat

Un risque est un événement aléatoire qui réduit la capacité de l'assureur à faire face à ses engagements. Dans le cadre des contrats épargne, de multiples risques doivent être quantifiés par l'assureur, on peut citer le risque viager, le risque de taux, de change et également le risque de rachat.

Un assuré peut, avant le terme de son contrat, retirer une partie ou l'intégralité du capital versé sur son contrat. C'est ce que l'on appelle un rachat partiel ou total de son épargne. Cette décision peut être prise à tout moment à la discrétion de l'assuré.

Pour l'assureur, il s'agit ici d'un risque à mesurer puisque cela lui impose de retirer une partie du capital qu'il a réinvesti. La valeur de rachat brute équivalant à la provision mathématique du contrat. Cependant il s'agit d'une définition à nuancer, dans le contexte actuel de taux bas l'assureur va préférer que ses assurés basculent sur des supports en unité de compte et portent eux-mêmes le risque de taux. Sur les fonds en euros, l'assureur portant le risque peut avoir du mal à trouver un rendement satisfaisant dans ce contexte actuel. Chose particulièrement problématique sur les contrats avec un taux minimal garanti, en plus de la nécessité de devoir immobiliser un certain capital. Dans cette situation, les rachats sur les fonds en euros peuvent être en quelque sorte un moindre mal.

Comme on a pu l'introduire précédemment, la fiscalité de l'assurance vie joue un rôle important sur le comportement de rachat et de sa temporalité. Le souscripteur va préférer racheter son contrat après 4 ou 8 ans d'ancienneté compte tenu du barème dégressif qui est appliqué.

On peut distinguer deux types de rachats :

- Le rachat structurel, lié aux besoins financiers de l'assuré, afin de financer ses besoins personnels ou tout simplement subvenir à un besoin de liquidité. Ce type de rachat est dépendant du comportement de l'assuré.

- Le rachat conjoncturel (ou dynamique), est lié aux conditions macroéconomiques. Plus spécifiquement, il peut s'agir par exemple d'un produit de la concurrence étant plus avantageux, l'assuré dans ce contexte rachètera son épargne pour la placer chez les concurrents. Ce type de rachat peut être lié à la performance du produit souscrit.

Depuis les années 1990, en raison de la croissance de l'assurance vie, les comportements liés aux rachats ont été de plus en plus étudiés. Aujourd'hui les résultats de ces études nous permettent de différencier plusieurs approches différentes pour ce risque, historiquement les 2 premières ont été privilégiées :

- Le rachat comme un besoin de liquidité (Outreville, 1990)
- Le rachat comme une opportunité d'arbitrage sur le marché (Pesando, 1974 et Cummins, 1975)
- Les études probabilistes : Le rachat y est étudié de façon individuelle, en calculant une probabilité de rachat pour chaque contrat. Modélisation du rachat par modèles linéaires généralisés (GLM) (Renshaw et Haberman, 1986), par régression logistique (Kim, 2005), par modèle Tobit (Lin, 2006) et modélisation du comportement de rachat par mélange GLM (Milhaud, 2012)
- Les études micro-économiques : On y étudie le rachat de manière individuelle en prenant en compte l'espérance d'utilité de l'assuré. (Mémoire de Fauvel et Le Pévédic, 2007)
- Les études financières : Dans ce cas, on considère le rachat comme une option à valoriser. (Shen et Huiping, 2004)
- Les études statistiques : Le rachat y est étudié de façon agrégée, des taux de rachats sont calculés par agrégation de portefeuille, c'est l'approche utilisée par CNP Assurances, où le calibrage des lois de rachats s'effectue par une modélisation des montants de rachats (partiels et totaux) en Best Estimate.

1.2. Approche retenue pour la modélisation du rachat

Dans un monde où la donnée prend une place de plus en plus importante au quotidien, lorsque l'on voit l'utilisation croissante de la *Data Science* qui est faite en matière de marketing, de rétention client ou encore d'optimisation des coûts, on se demande s'il ne serait pas envisageable d'étendre ce champ de compétences à l'activité d'actuariat.

Etant dans un service de *Data Science* en recherche et développement actuarielle, nous sommes amenés à effectuer régulièrement des études sur les lois d'expérience, dont les lois comportementales de rachats. Une des interrogations du service s'est portée sur l'apport d'une approche par *Machine Learning* à l'estimation du risque du rachat par rapport à la modélisation classique de ces lois.

L'approche retenue consiste à estimer de manière individuelle la période où l'assuré serait le plus susceptible de racheter son contrat. En d'autres termes, elle vise à prédire pour chaque contrat, le moment où l'assuré pourrait effectuer un rachat total. Cela permet de dégager deux cas d'utilisation :

- Ciblage client : Disposer de la date du rachat probable permettrait de contacter le détenteur des capitaux afin de lui proposer de réinvestir ses fonds, dans un optique de rétention de la clientèle.

- Gestion des allocations d'actifs : Lorsqu'un contrat est souscrit, les fonds de ce contrat sont investis par la compagnie d'assurance dans des actifs, permettant ainsi de dégager un rendement. Cependant la nature de l'allocation dépend de la durée de conservation des capitaux, et le rachat constitue le risque principal de sortie de capital du contrat. Identifier la temporalité du rachat de manière individuelle permettrait d'affiner la

connaissance des flux de trésorerie futurs, et entrerait dans le cadre de l'estimation en Best Estimate de la directive Solvabilité 2.

Notre approche tend donc à répondre sur la faisabilité d'un tel projet via une approche *Machine Learning*.

Tout d'abord, il sera question d'effectuer cette étude sur la base des informations disponibles à la souscription du contrat uniquement. Et dans un second temps, nous élargirons l'étude pour pouvoir prendre en compte les données de la vie du contrat.

Chapitre

2

Construction de la base de données et analyse descriptive

La base de données utilisée dans le cadre de cette étude a été constituée à partir d'un sous-portefeuille de contrats de CNP Assurances. Le périmètre étudié est celui des contrats d'assurance vie en épargne, composé de contrats souscrits entre janvier 2000 et décembre 2019. Les rachats partiels sont exclus de l'étude du fait du côté aléatoire du montant racheté. Ainsi notre étude se porte sur les rachats totaux, le capital retiré étant connu par avance du fait de la valorisation du contrat à la date du rachat.

2.1. Présentation des données utilisées

Une compagnie d'assurances dispose de diverses données lui permettant d'avoir une meilleure compréhension des risques qu'elle supporte. Ces données peuvent être internes (issues des systèmes de gestion car propres à l'assureur), mais aussi externes (open data ou autres).

En pratique, l'exploitation des données fait face à divers problèmes, on peut citer :

- La qualité des données est le frein principal à leur exploitation, on peut être confronté à des soucis d'exhaustivité, de cohérence, de pertinence ou encore d'exactitude.
- La disponibilité et le traitement des données, pour des raisons techniques (limitations pour le stockage et le traitement des données), ou des raisons réglementaires (ex : RGPD).

Les données utilisées par les compagnies d'assurance pour mesurer les risques comportementaux sont dans un premier temps les informations brutes recueillies à la souscription du contrat (informations sur l'assuré et données du contrat), et par la suite les données d'inventaire (ex : encours, mouvements sur contrat, etc…)

Tout d'abord, nous n'exploiterons que les données disponibles à la date de souscription pour évaluer notre approche sans disposer d'historique. Dans un second temps, nous élargirons notre

modélisation en ajoutant les données d'inventaire, disposant ainsi des informations sur les mouvements effectués pendant la vie du contrat.

Dans les parties suivantes, nous expliciterons les différentes extractions et traitements de données réalisés dans le but d'obtenir notre base d'étude finale. Tous ces traitements ont été effectués sous Python, sur des serveurs dédiés disposant de 800Go de RAM, permettant ainsi une manipulation des données sans contraintes computationnelles.

2.1.1. Extraction et retraitement des données

Les données utilisées proviennent de différentes entités et sont par la suite regroupées dans un même système de données accessible en interne. Dans ce système on peut sélectionner les données souhaitées et effectuer des requêtes afin d'en extraire le contenu.

Afin de pourvoir constituer notre base d'étude, il nous faut extraire les informations sur les contrats du périmètre épargne, les informations sur les détenteurs de ces contrats d'assurance, mais également les informations sur les événements associés à ce contrat. Cela constituera notre base de travail pour la suite de l'étude.

Nous avons dans un premier temps effectué une analyse détaillée de la base de données, en étudiant chaque table disponible. Ce premier aperçu nous a permis de sélectionner des données qui étaient satisfaisantes par rapport à nos exigences, à savoir des données pertinentes, exhaustives et fiables, nous permettant de créer une base à la maille Client & Contrat pour y étudier les comportements de rachats.

Le contrôle des données est un point crucial de toute étude actuarielle. Les données doivent être exhaustives, car il faut ici pouvoir justifier d'un historique et de volume suffisant pour que

l'information contenue dans la donnée soit exploitable. Pour la fiabilité, il s'agit de contrôler la cohérence pour s'assurer que les données utilisées sont dépourvues d'erreur, ou a minima, qu'elles disposent d'un niveau de confiance élevé, on distingue deux types de contrôles :

- Les contrôles « *a priori* » : ces contrôles de premier rang permettent l'identification de modifications qu'aurait connu l'assureur dans la collecte de ses données (évolution ou changement du processus de gestion, fusion d'entités…) dans le but de réduire le risque d'introduire des troncatures dans les données utilisées, ou d'identifier ces dernières.

- Les contrôles de cohérence interne à la base de données : l'objectif est de contrôler la cohérence entre les éléments communiqués dans la base de données. Il s'agit de repérer certaines situations invraisemblables.

Une grande importance a été apportée au nettoyage des données, celles qui sont extraites sont dans la plupart des cas sous un format brut et nécessitent donc d'être retraitées.

2.1.1.1. Table des produits

Il s'agit d'une table de quelques milliers de lignes où l'on trouve les informations concernant les produits commercialisés par CNP Assurances, on peut citer :

- Le code du produit
- Le code PAPIV (le code permettant d'identifier la famille de produit à laquelle il appartient)
- Le code de portefeuille du produit
- L'unité du support sur lequel le produit est commercialisé (contrats € ou UC)
- Le code du partenaire commercial qui dispose de ce produit
- Le cadre fiscal auquel il est rattaché

De cette table, nous filtrons les données à partir d'un référentiel interne afin de garder les produits du périmètre épargne.

2.1.1.2. Table des contrats

Il s'agit d'une table de plusieurs dizaines de millions de lignes, où l'on trouve les informations concernant les termes du contrat souscrit. On dispose du :

- Numéro du contrat
- Code produit commercialisé (faisant le lien avec la table produit)
- Code d'entrée du contrat (s'il s'agit d'une affaire nouvelle, d'un transfert Fourgous, etc…)
- Date de souscription du contrat
- Code de sortie du contrat (nul si le contrat est toujours actif)
- Date de sortie du contrat

A partir des différents codes de sortie, on peut identifier les contrats clôturés pour cause de rachat total, de sinistre, de transfert, etc…

2.1.1.3. Table des rôles

Il s'agit ici d'une table de plus d'une dizaine de millions de lignes, c'est une table de jointure entre les informations des assurés et celle des contrats. On dispose de l'identifiant de l'assuré, du numéro du contrat et d'un code rôle.

Cette dernière variable prend la forme d'un code de plusieurs caractères binaires, et nous permet d'identifier la position de la personne vis-à-vis du contrat (Souscripteur, Assuré, Bénéficiaire,

Coassuré, etc…). Pour notre étude nous avons sélectionné uniquement les assurés, propriétaires de leur contrat.

Nous avons également identifié les contrats en situation de co-assurance, ce qui se traduit par une duplication des lignes de contrats pour les deux coassurés lors de la jointure des tables.

2.1.1.4. Table des clients

Cette table regroupe toutes les informations sur les assurés de CNP Assurances et compte plusieurs dizaines de millions de lignes de données. C'est cette table qui nous permet de récupérer les informations personnelles des clients, on peut citer :

- Date de naissance
- Code sexe
- Situation familiale
- Nationalité
- Domicile

2.1.1.5. Table des événements de souscriptions

Cette table a été construite à partir de la table des événements sur contrat, il s'agit d'une table regroupant toutes les informations des différents flux effectués sur le contrat depuis sa création.

Nous avons commencé par identifier les codes d'événements correspondant aux versements de souscriptions, pour ensuite extraire ces informations pour chaque contrat. Cette table ainsi constituée nous donne les informations sur la date de versement, la nature du support sur lequel à été fait le versement (€/UC), le montant du versement, ainsi que le type de versement souscrit (versement prime unique, régulière, libre, etc…).

Cette table disposant d'une ligne par versement, nous avons regroupé les informations par contrat et par date, nous permettant d'identifier les versements sur des contrats multisupports, et agrégant les montants de versements à la souscription. Ainsi construite, cette table comptabilise plus d'une dizaine de millions de lignes.

2.1.1.6. Table des événements de rachats

Comme précédemment et à partir de la table des événements, on a identifié les flux correspondant à des rachats totaux et extrait les données concernées.

On a remarqué que pour certains contrats, des demandes de rachats avaient été effectuées puis annulées.

Lorsqu'une demande de rachat est effectuée, un flux sortant est créé dans la base de données. Cette demande peut être annulée par l'assuré ce qui constitue une annulation de la demande de rachat. Une ligne contenant un flux entrant est créée le même jour afin d'annuler la demande de rachat initiale.

Ainsi, il est possible pour ces contrats d'observer plusieurs dates pour des flux de rachat total. Mais seul le flux le plus récent peut être considéré comme tel, car non annulé. On a ainsi décidé de sélectionner systématiquement la date de rachat totale la plus récente pour chaque contrat, correspondant à la date de clôture effective de ce dernier.

2.1.2. Agrégation des données

Après avoir sélectionné et extrait ces tables, nous les avons donc jointes entre-elles sur la base du numéro de contrat et de l'identifiant client pour constituer un groupe de données cohérent.

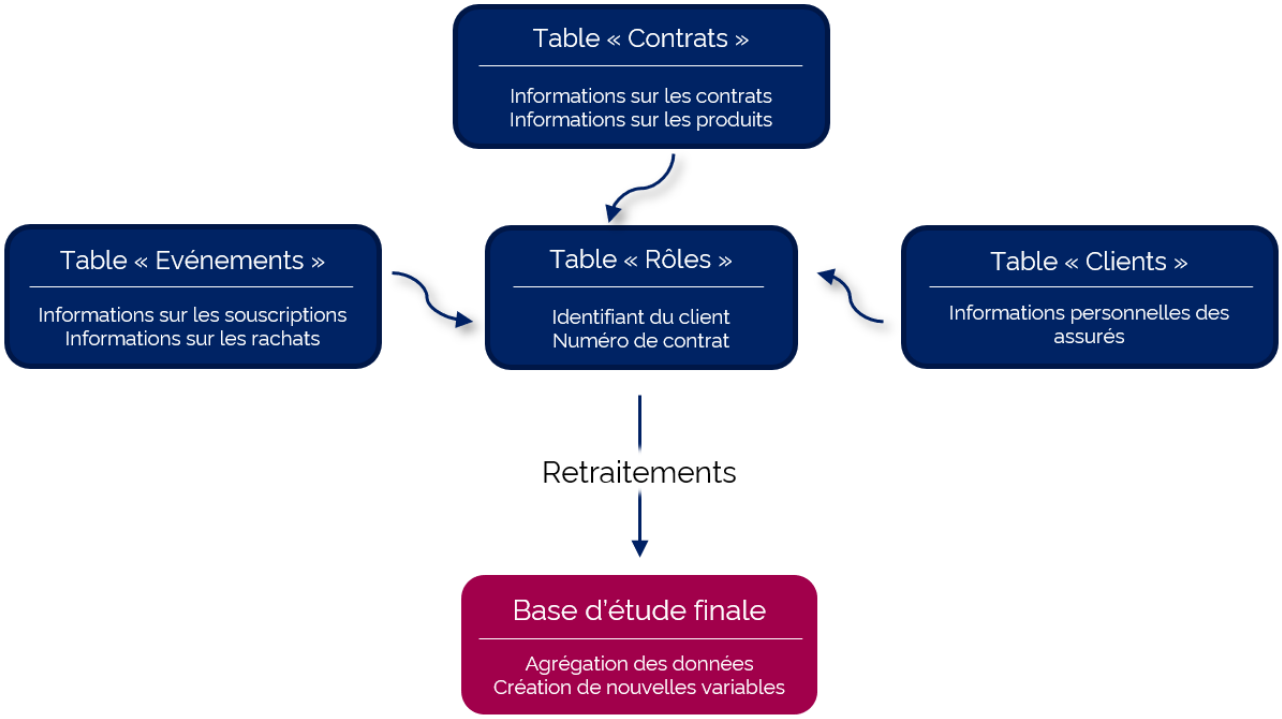

Illustration 2 : Schéma d'agrégation des tables

Comme nous pouvons le voir sur le schéma précédent, les données que nous avons extraites sont agrégées à la maille Client et Contrat, il faut comprendre par là que si un contrat à été souscrit en co-assurance, alors la ligne du contrat correspondant sera présente deux fois, avec les informations concernant les deux individus.

A cette étape nous avons remarqué des conflits d'exactitude entre certaines de nos variables.

Nous avons également porté un grand intérêt quant à la cohérence des données et supprimé toutes les lignes où :

- La date de naissance n'est pas renseignée
- La date de naissance renseignée est supérieure à la date de souscription
- La date de souscription est supérieure à la date de rachat total où à la date de sortie du contrat
- La date de décès est inférieure à la date de souscription
- Le contrat est indiqué comme étant clos, mais la date de sortie n'est pas renseignée

Enfin, nous supprimons également toutes les lignes où l'on observe une cause de sortie précoce du portefeuille, à savoir un type de sortie qui survient majoritairement dans les 6 premiers mois de la vie du contrat. On a pu identifier ces causes de sorties et supprimer les contrats concernés : par exemple les sorties pour cause de résiliation, renonciation, première prime impayée, etc….

Ces retraitements correspondent à une suppression de 0.04% des observations de la base. Un retraitement supplémentaire n'est pas nécessaire dans la mesure où l'on dispose d'une grande quantité de données de bonne qualité dans la table retraitée.

2.1.2.1. Phénomène de censure

Nous rappelons que notre étude porte sur la prédiction d'une durée jusqu'à un événement, le rachat total. C'est une problématique qui entre dans le domaine des modèles de survie, ces modèles se distinguent d'un sujet de régression classique. En effet l'élément que l'on cherche à prédire est une durée, par conséquent c'est une variable qui est :

- Positive : Il faut comprendre que la durée est toujours positive, dans le sens où une durée négative pour atteindre un événement est un non-sens.

- Censurée : Il est également possible que la durée soit inconnue, car au moment de la prise d'information, l'événement étudié n'a pas été observé, c'est ce que l'on appelle le phénomène de censure.

Ce second point est important puisqu'il nous prive d'une partie de l'information que l'on cherche à mesurer.

Pour mieux comprendre prenons l'exemple suivant :

N° Contrat	…	Date souscription	Date sortie	Type sortie	Durée jusqu'au rachat (en mois)
1	…	01/01/2017	01/01/2020	Rachat Total	36
2	…	15/09/2018	Inconnu	Non Sorti	[27.5, +inf[
3		03/04/2019	10/02/2020	Rachat Total	14.2
4	…	18/02/2016	24/07/2019	Sinistre-Décès	[41.2, +inf[
5	…	27/06/2015	14/08/2018	Rachat Total	37.6

En observant les données utilisées dans le cadre de notre étude, on est en mesure d'identifier trois catégories d'observation :

- Les contrats résultant en un rachat.
- Les sorties du portefeuille pour des causes diverses, autre qu'un rachat (sinistre, transfert, etc…).
- Les contrats toujours actifs.

Pour les lignes 1, 3 et 5, qui se trouvent être des cas de rachat total, on dispose d'une valeur unique définissant la durée. Puisque la date de l'événement de rachat est connue, l'information est complète.

Sur la ligne 2, la durée est un intervalle de temps et non une valeur unique, on est en présence d'un cas de durée censurée. La raison est qu'à la date d'observation, ce contrat est toujours actif et donc l'événement de rachat n'a pas encore été observé. Par conséquent la durée jusqu'au rachat de ce contrat est une durée qui est nécessairement plus grande que celle dont on dispose aujourd'hui (dans l'exemple 27.5 mois, durée écoulée à la date d'observation). Ce type de censure est une censure non-informative car indépendante de l'événement ciblé.

Sur la ligne 4, l'individu a eu un sinistre et son contrat a été clôturé, Il s'agit également d'un cas de censure puisque l'événement de rachat n'est pas observé. Hypothétiquement, l'individu aurait très bien pu racheter son contrat dans un futur plus ou moins éloigné s'il n'avait pas eu son sinistre. La seule information dont on dispose dans ce cas, c'est que le sinistre survient avant l'événement de rachat total. Ainsi la durée présumée jusqu'au rachat total est supérieure à la durée de vie du contrat. Il s'agit également d'une censure non-informative.

Il existe deux types de censure :

- La censure à droite, où la valeur minimale est inconnue.
- La censure à gauche, où la valeur maximale est inconnue.

Dans notre étude, la date de souscription est renseignée pour toutes nos données, ce qui nous permet de quantifier une durée de vie du contrat peu importe la situation, ce qui élimine les cas de censure à droite. En revanche, les cas de censure gauche sont ceux que l'on va le plus fréquemment rencontrer. La date de rachat n'étant renseignée que lorsqu'il y a eu un événement de rachat, cela signifie que pour les autres données cette information n'est pas disponible.

Notre étude prévoit d'inclure cette problématique par le biais d'un modèle d'apprentissage automatique permettant l'utilisation de données censurées.

2.1.2.2. Troncature

On notera également la présence d'un autre type de phénomène dans l'étude des durées de survie, il s'agit de la troncature. Le phénomène de troncature se définit comme la possibilité d'observer une valeur uniquement si celle-ci appartient à un sous-ensemble de valeurs possibles.

Dans notre cas, les contrats souscrits avant le début de cette période ne sont pas pris en compte et les événements survenus après la fin de la période d'étude ne sont pas considérés non plus. Ce qui signifie que la durée maximale jusqu'au rachat est tronquée par la durée de notre période d'étude choisie.

2.1.2.3. Périodicité et segmentation retenue

Les données extraites nous permettent d'observer les rachats totaux sur la période 2011-2021, les contrats ayant une date de sortie antérieure à 2010 étant pour la plupart historisés et indisponibles, on observe une information incomplète sur les rachats. Dans le cadre de notre étude, nous avons donc sélectionné tous les contrats actifs à date du 1er janvier 2011.

La crise du Covid-19 ayant impacté la stabilité des marchés financiers en 2020 (Ill. 3 suivante), on ne peut pas écarter l'hypothèse d'un impact du Covid sur les comportements de rachat en 2020, c'est pour cela que dans le cadre de notre étude nous avons décidé de ne pas prendre en compte cette année et de tronquer notre base au 1er janvier 2020.

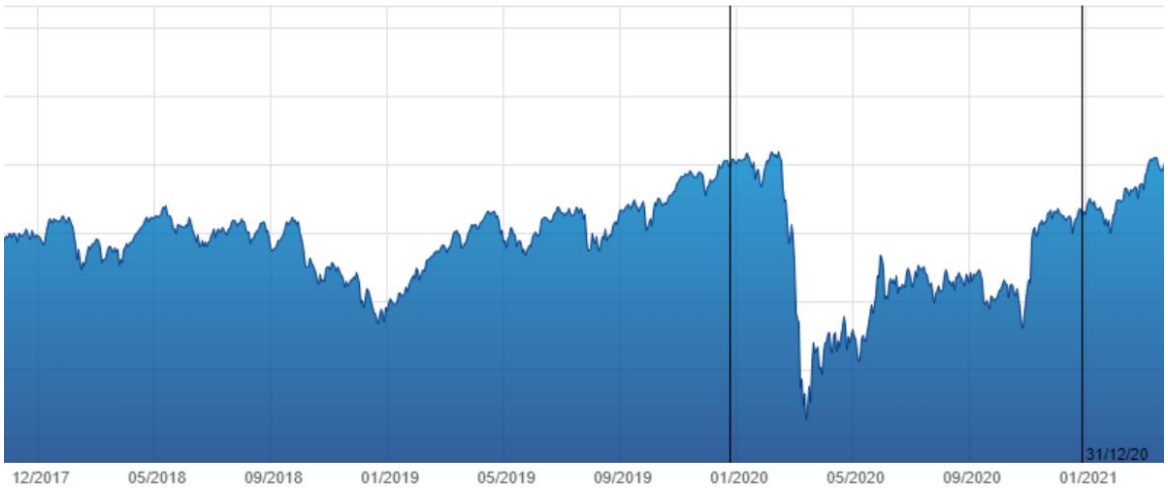

Illustration 3 : Evolution de l'indice du CAC40 (l'année 2020 se situe entre les repères verticaux)

Le graphique ci-dessous représente la représentation de nos réseaux de distribution dans notre jeu de données. On remarque que les partenaires de la catégorie « Autres réseaux » représentent une part négligeable. On a donc décidé de segmenter notre étude sur les produits épargne commercialisés par les partenaires nommés « A » et « B ».

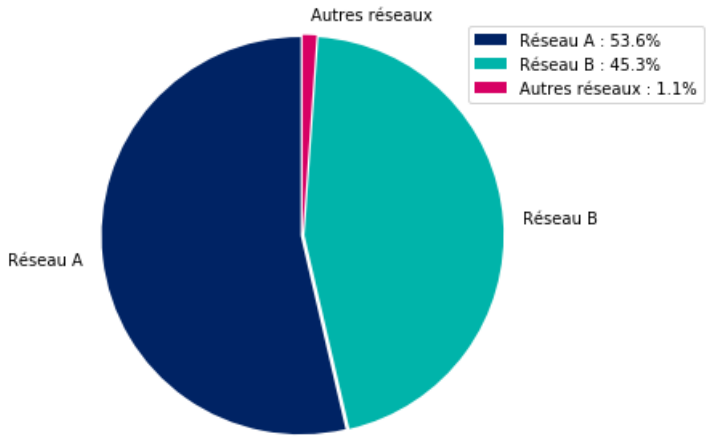

Illustration 4 : Répartition des contrats par réseau de distribution

2.1.3. Création des features

Dans cette section, nous emploierons le terme *features* pour désigner les variables explicatives de nos modèles.

La création de nouvelles *features*, communément appelé *feature engineering*, consiste à créer de nouvelles caractéristiques à partir des données brutes. Plus précisément, il s'agit de composer avec les données existantes afin de les rendre plus pertinentes pour un algorithme de M*achine Learning* prédictif.

Ainsi, en plus des variables extraites auparavant, nous avons décidé de créer les *features* suivantes :

- Age_Souscription : L'âge de l'assuré lors de la souscription du contrat.
- Nb_Contrats_Actifs : Le nombre de contrats épargne actifs de l'assuré à la date de souscription du contrat.
- Coassurance : Variable booléenne qui indique si le contrat est en co-souscription ou non.
- Anc_Fiscale : Variable booléenne indiquant si le contrat à été souscrit via un transfert de produit épargne, permettant à son souscripteur de garder l'ancienneté fiscale de son ancien contrat.

Nous définissions notre variable cible, $TARGET$, qui correspond à la durée avant le rachat total, en incorporant également dans le calcul la durée jusqu'à la censure pour les cas des données censurées, ces durées sont exprimées en mois :

$$TARGET = \begin{cases} Date\ de\ rachat - Date\ de\ souscription, & si\ Rachat = 1 \\ Date\ de\ sortie - Date\ de\ souscription, & si\ Sortie = 1 \\ Date\ de\ troncature - Date\ de\ souscription, & si\ Sortie = 0 \end{cases}$$

Cette variable cible correspond à la durée minimale d'observation du contrat dans nos bases, avec deux cas de figure :

- Le contrat est toujours actif à la date de troncature de notre base (01/01/2020), dans ce cas on observe aucune sortie et la durée correspond à celle entre la date de souscription et le 01/01/2020.

- Si l'individu sort avant la date de troncature, alors dans ce cas les mouvements du contrat ne seront plus observés après sa date de sortie du portefeuille, et la durée correspondra à la différence entre la date de souscription et la date de sortie du contrat.

Nous définissons aussi une cible secondaire et simplifiée, ne traitant que les cas de données non censurées, elle est définie comme suit :

$$TARGET_{hors\ censure} = \begin{cases} Date\ de\ rachat - Date\ de\ souscription, & si\ Rachat = 1 \\ Null\ , & si\ Rachat = 0 \end{cases}$$

2.2. Statistiques descriptives

Avant de passer à l'étape de modélisation, il est nécessaire d'effectuer une analyse descriptive de nos données. Par un souci de confidentialité des données, aucune valeur numérique ne sera indiquée en ordonnée dans la suite de ce document.

Pour rappel, notre base d'étude comporte les contrats d'épargne actifs ou rachetés sur la période 2011-2019 issus d'un portefeuille de contrat souscrits depuis 2000.

A date de la troncature au 31/12/2019, notre base est constituée de :

- 69.5% de contrats toujours en activité
- 19.7% de contrats rachetés en totalité
- 10.8% de contrats clôturés pour des causes annexes (sinistre, transfert, etc…)

Afin de pouvoir comparer nos résultats par profils de risque nous avons utilisé des regroupements de produits d'assurance vie. Ces regroupements que nous appellerons GRH (Groupes de Risques Homogènes) sont créés sur la base d'une expertise métiers des contrats, on distingue :

- Les contrats « Haut de Gamme »
- Les contrats « Grand Public Agés »
- Les contrats « Grand Public Jeunes »

Illustration 5 : Nombre de souscriptions par année

Le graphe ci-dessus représente le nombre de souscriptions par année en fonction des produits composant les GRH entre 2000 et 2019.

On observe une tendance à la baisse du nombre de souscriptions depuis l'année 2011 ainsi qu'une chute importante du nombre de collecte depuis l'année 2017. Cette tendance est imputable à une baisse du nombre de souscriptions pour un des produits les plus commercialisés de notre portefeuille de contrats faisant partie du groupe de contrats « Grand Public Agé ».

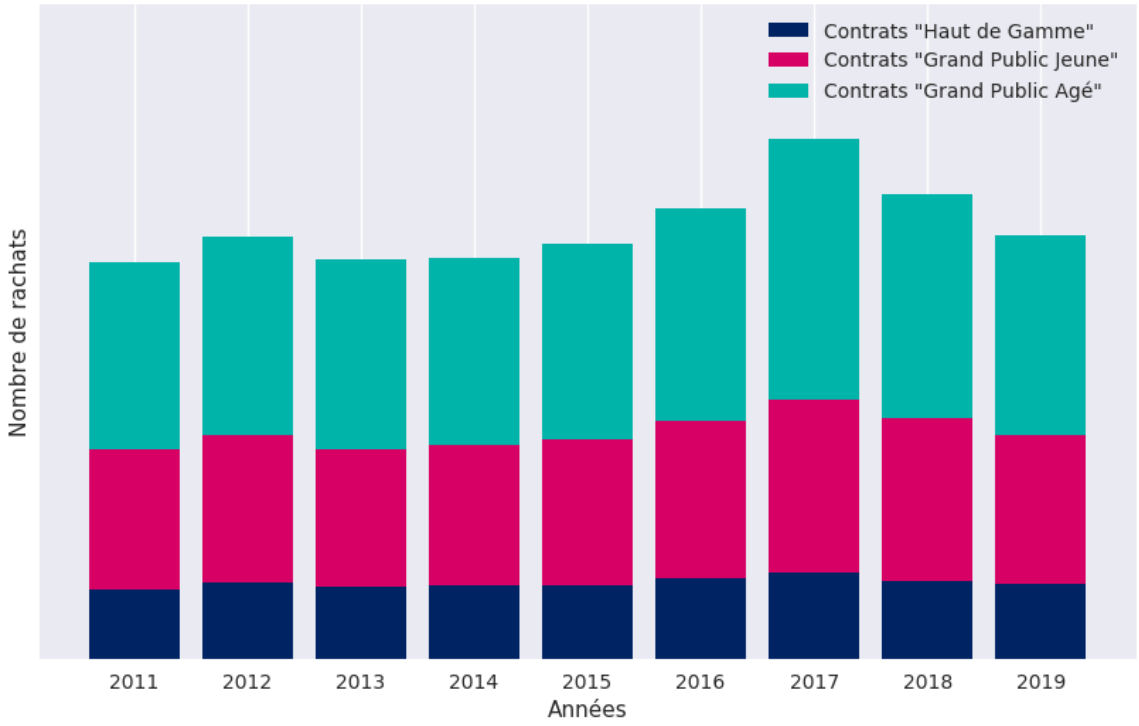

Illustration 6 : Nombre de rachats totaux par année

Ce graphique représente le nombre de rachats totaux effectués par année entre 2011 et 2019 en fonction des produits composant les GRH.

On remarque un léger rebond du nombre de rachats en 2017. On peut supposer que cela est causé par le rachat des grands nombres de contrats « Grand Public Agé » souscrits en 2009 et arrivant à leur 8ème année d'ancienneté, où la fiscalité devient plus souple.

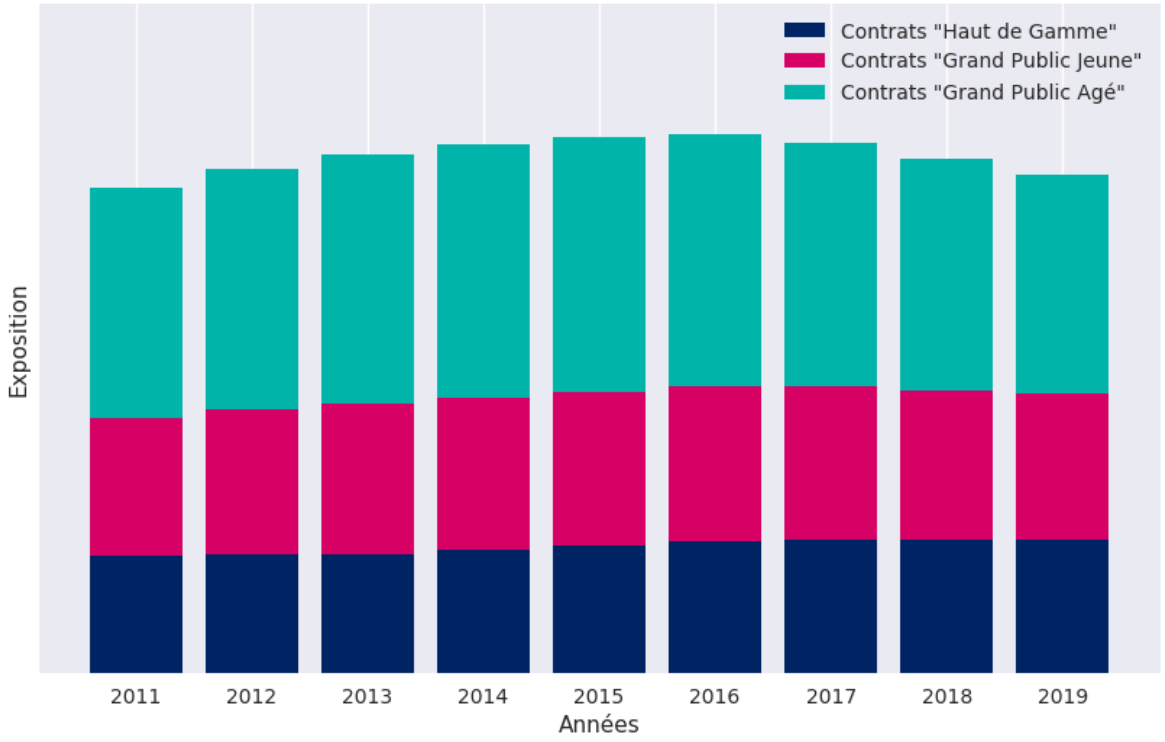

Illustration 7 : Exposition par année

Ce graphique représente l'exposition au risque des contrats composant les différents GRH sur la période 2011 à 2019. L'exposition est ici calculée à l'aide de l'estimateur binomial. On considère que toutes les entrées ou sorties du portefeuille pendant l'année s'effectuent au milieu de celle-ci.

Le résultat observé ici est déductible des deux graphiques précédents. La baisse drastique de la collecte depuis 2017 associée à un nombre de rachat en légère augmentation sur cette même période par rapport aux années précédentes donne lieu à une légère baisse de l'exposition globale depuis 2017, en particulier pour les contrats « Grand Public ».

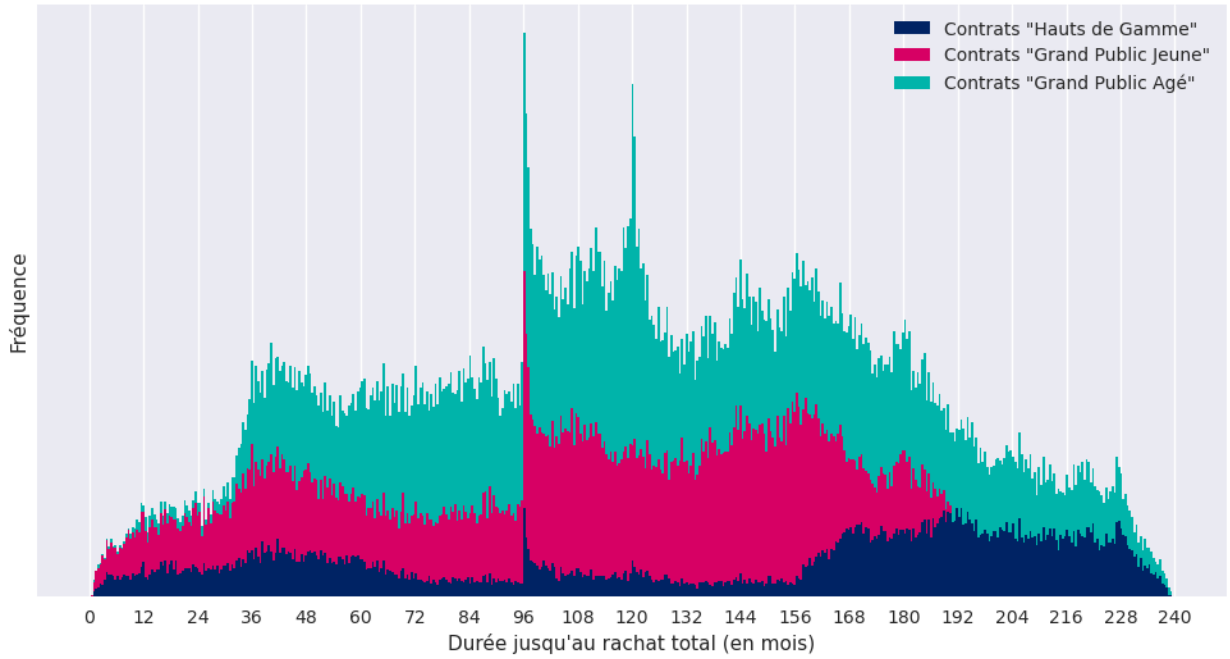

Illustration 8 : Histogramme de la durée jusqu'au rachat en fonction du GRH

Le graphique ci-contre représente la distribution de la durée séparant la date de souscription et la date du rachat total dans notre jeu de données. Cette durée représente notre cible que l'on cherche à modéliser.

On observe un net pic de rachat au 96ème mois (la 8ème année) pour tous les types de contrats. Cela s'explique par l'allégement fiscal qui a lieu lorsque que le contrat atteint ce niveau d'ancienneté. On remarque également un deuxième pic de rachats lors du 10ème anniversaire du contrat pour les produits « Grand Public Agé », cela provient du comportement de rachat que l'on observe pour un unique produit.

Pour les contrats dits « Haut de Gamme », on remarque une grosse proportion de contrats rachetés après 13 ans d'ancienneté (156ème mois). Cela vient du fait que la plupart de ces contrats ont été souscrits dans le début des années 2000 et nous observons les rachats de 2011 à 2019.

Ainsi les contrats « Haut de Gamme » comportent une grande proportion de contrats d'ancienneté élevée par rapport au reste de notre base, ce qui influe sur les proportions de rachats sur les durées les plus grandes.

Un constat similaire peut être donné en ce qui concerne les contrats « Grand Public Jeune », ceux-ci ont commencé à être commercialisés à partir du milieu des années 2000, il est donc naturel de ne pas observer de rachat au-delà de la 16ème année puisqu'aucun de ces contrats n'a atteint cette d'ancienneté.

Illustration 9 : Répartition de la durée jusqu'au rachat en fonction de l'année de rachat

Le graphique ci-dessus représente la répartition des valeurs extrêmes, des quantiles et de la médiane des valeurs de la variable cible selon l'année du rachat.

On remarque une augmentation continue de la durée jusqu'au rachat d'année en année. Ce qui est tout à fait cohérent puisque les premiers contrats souscrits dans notre portefeuille datent de l'an 2000, à mesure que les années passent, à moins que les contrats les plus anciens soient tous rachetés, l'ancienneté moyenne des contrats composant notre portefeuille augmentera.

Un moyen de restreindre ce phénomène serait d'étudier la distribution des durées en prenant un intervalle de temps fixé pour chaque année. C'est ce que nous allons observer par la suite.

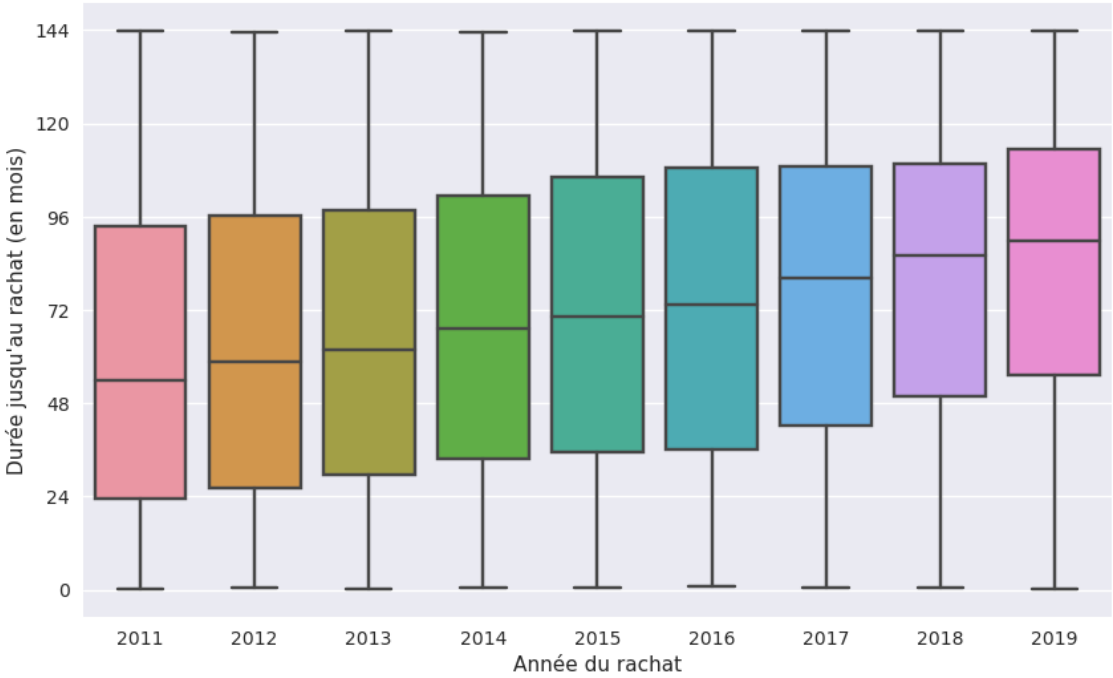

Illustration 10 : Répartition de la variable cible en fonction de l'année de rachat (période coulissante)

Le graphique ci-dessus représente la distribution des valeurs prises par la variable cible selon l'année du rachat. A la différence du graphique précédent, les rachats observés ici ont été restreints à une observation de 11 ans (Des contrats rachetés en 2011 ayant été souscrits après l'an 2000, aux contrats rachetés en 2019 ayant été souscrits à partir de 2008). Cela nous permet d'étudier les durées jusqu'au rachat sur la base d'un intervalle temporel équivalent pour chaque année.

On observe toujours une augmentation de la durée jusqu'au rachat en fonction de l'année, mais celle-ci reste plus maitrisée que l'accroissement aperçu dans le graphique précédent. Cette hausse annuelle trouve un élément d'explication dans l'inégalité du nombre de contrats souscrits annuellement. En effet, la collecte de contrats était plus importante avant 2010, avant de chuter progressivement. A mesure que le temps passe, les anciens contrats font vieillir l'ancienneté moyenne du portefeuille, car il y en a plus en proportion.

Dans nos modélisations, c'est cette approche que nous privilégierons pour permettre de contenir ce biais lié à l'évolution de l'ancienneté des contrats.

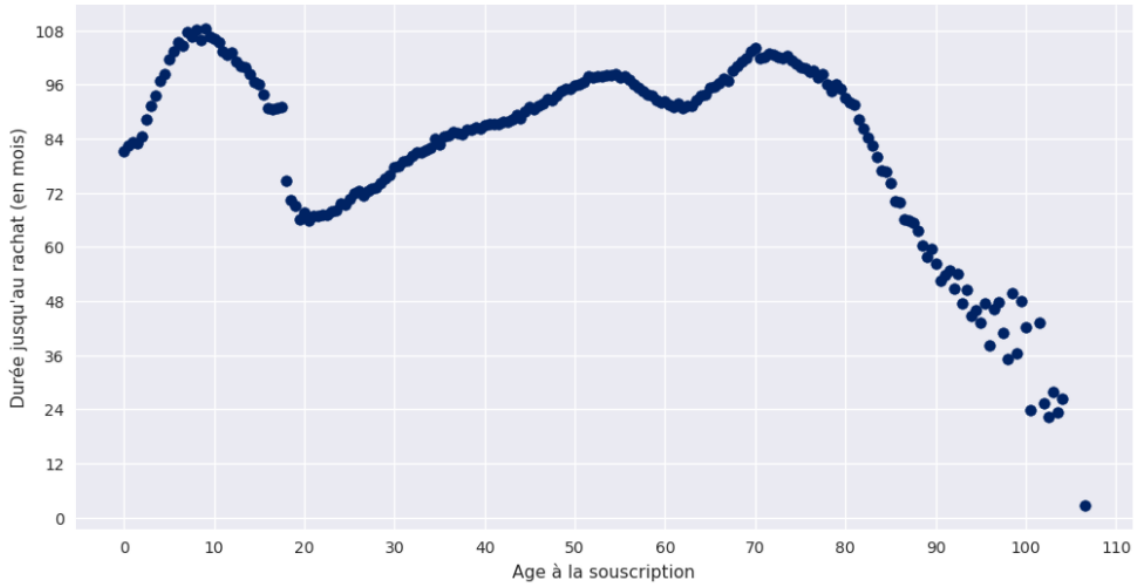

Illustration 11 : Durée moyenne jusqu'au rachat en fonction de l'âge à la souscription

Ce graphique est un nuage de points représentant la durée moyenne jusqu'au rachat (sur l'axe des ordonnées) en fonction de l'âge à la souscription. Les âges étant calculés par année en utilisant l'arrondi au cinquième près.

Ce graphique est intéressant puisqu'il nous permet de faire un lien visible entre l'âge du souscripteur et la durée pendant laquelle il va conserver son contrat. Cette variable explicative servira par la suite à l'entrainement des modèles.

On observe que pour les enfants en bas-âge (ou plutôt les membres de leur famille qui souscrivent en leur nom), ceux-ci ont tendance à conserver leur contrat au-delà de la 8ème année

d'ancienneté. Puis, pour les contrats souscrits par les jeunes actifs, on observe une baisse de la durée moyenne ce qui coïncide avec le besoin en liquidités durant cet âge (hypothèse de nécessité urgente de ressources pour l'assuré, Outreville, 1990). La durée augmente progressivement jusqu'aux contrats souscrits à l'âge de la retraite, où la durée descend subitement. Enfin, vers les âges élevés, elle remonte jusqu'à l'âge de 70 ans, où l'effet de la mortalité pèse sur le rachat et l'on observe une baisse drastique de la durée moyenne.

2.3. Descriptif du jeu de données final

A l'issue des étapes de retraitement des données, nous disposons des variables explicatives suivantes :

Variables	Descriptif	Type de donnée
Coassurance	Contrat souscrit en coassurance	Binaire : 0 ou 1
Code_Nationalité	Nationalité du souscripteur	String : Choix multiples
Code_Sexe	Sexe du souscripteur	Binaire : 0 ou 1
Situation_Familiale	Situation familiale du souscripteur	String : Choix multiples
Domicile	Région du domicile du souscripteur	String : Choix multiples
Code_PAPIV	Code de regroupement du produit	String : Choix multiples
Cadre_Fiscal	Cadre fiscal du produit	String : Choix multiples
Code_Réseau	Réseau de distribution du produit	Binaire : 0 ou 1
Entrée_Contrat	Type d'entrée dans le contrat	String : Choix multiples
Antériorité_Fiscale	Antériorité fiscale du contrat	Binaire : 0 ou 1
Age_Souscription	Age à la souscription	Numérique (Positif)
Versement_Initial	Versement à la souscription	Numérique (Positif)
Nb_Contrats_Actifs	Nombre de contrats actif lors de la souscription	Numérique (Positif)
EUR	Souscription à un contrat en euros	Binaire : 0 ou 1
UC	Souscription à un contrat en UC	Binaire : 0 ou 1
SOUS_PL	Souscription à des primes libres	Binaire : 0 ou 1
SOUS_PU	Souscriptions à des primes uniques	Binaire : 0 ou 1

Afin d'affiner notre compréhension entre nos variables explicatives et notre cible de durée, nous décidons d'observer les corrélations de nos variables.

Puisque celles-ci sont majoritairement qualitatives ou binaires nous allons à l'aide du test de V de Cramer pour observer l'indépendance, ou non, de nos variables. Pour les variables continues, nous avons arrondi les modalités afin de rendre les valeurs discrètes et ainsi pouvoir être comparées.

Le V de Cramer se calcule comme suit :

$$V = \sqrt{\frac{\chi^2}{n * DLL}}$$

Où χ^2 représente la valeur du test de khi² d'indépendance, n l'effectif, et DLL le degré de liberté.

Plus V est proche de zéro, plus il y a indépendance entre les variables étudiées. Entre 0.2 et 0.4 on considérera que la relation entre les variables est modérée, au-delà et jusqu'à 0.8 la liaison est forte, et lorsque V tend vers 1 il y a colinéarité entre les variables.

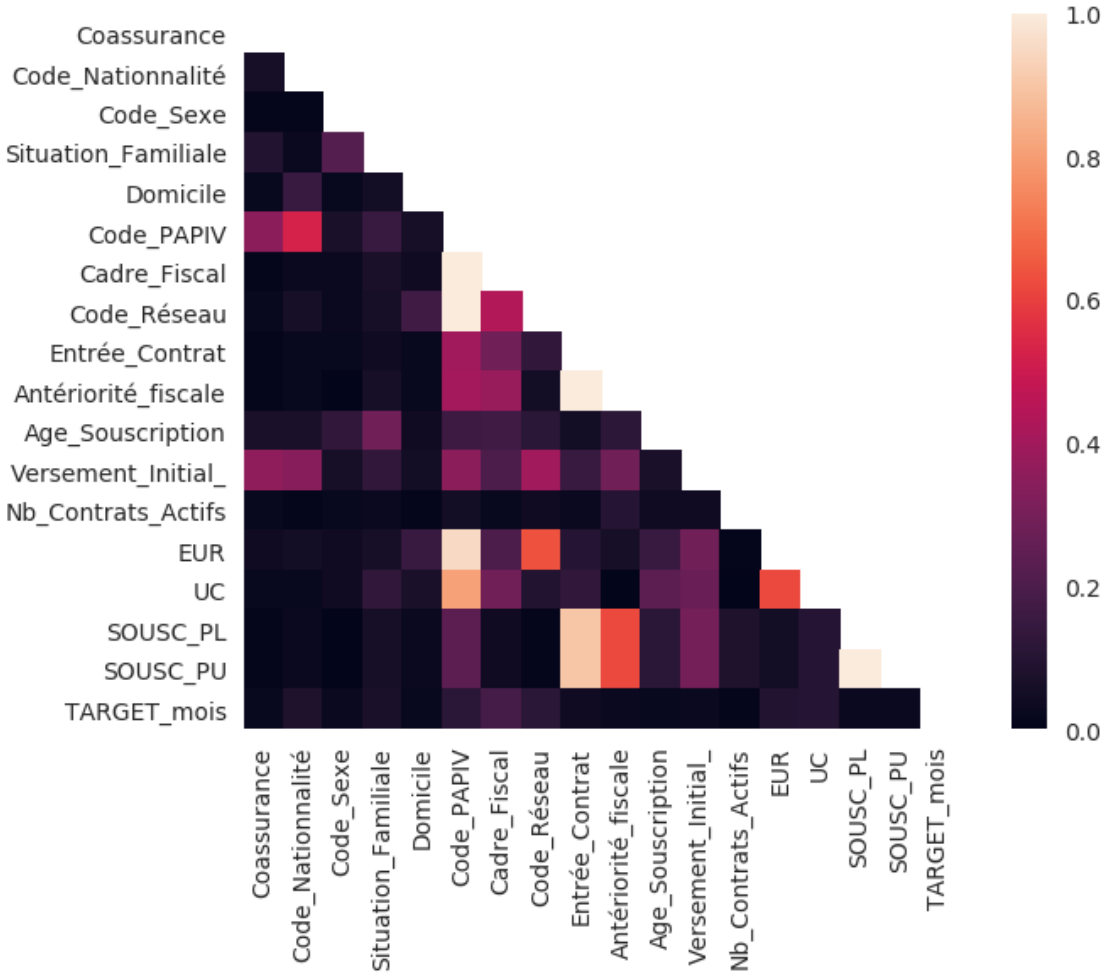

Illustration 12 : Matrice de corrélation des variables explicatives

Le graphique ci-dessus représente la dépendance de nos variables entre-elles calculé à l'aide du V de Cramer, sur les données présentant un cas de rachat.

On observe une dépendance modérée à forte entre les codes PAPIV (regroupement de codes produits) et les types de supports souscrits, le cadre fiscal et le code réseau. Ce qui est cohérent puisque ces informations ont été obtenues en fonction du produit commercialisé.

On note également la présence d'une forte corrélation entre le type d'entrée dans le contrat et l'antériorité fiscale, ainsi que le type de versement souscrit (SOUS_PL et SOUS_ PU). De la même manière, ces dernières informations ont été recueillies sur la base de l'entrée dans le contrat. Il est donc cohérent d'observer cette dépendance.

D'une manière générale, les variables que nous avons retenues sont très faiblement corrélées avec notre variable cible tout en étant relativement peu corrélées entre-elles à quelques exceptions près. Cela signifie que nos variables explicatives apportent une information différente pour notre modélisation.

Chapitre 3

Approche n°1 : Modélisation par régression

3.1. Introduction au Machine Learning

L'approche *Machine Learning* est une méthode où l'on cherche à prédire une valeur à l'aide de données d'apprentissage. On juge le plus souvent la qualité du modèle en fonction de l'adéquation des prédictions à la réalité, sur un jeu de données indépendant de celui servant à l'apprentissage. L'apprentissage lui, est basé sur un principe de minimisation du risque empirique.

Il existe plusieurs familles de *Machine Learning*, on peut citer :

- Les algorithmes d'apprentissage supervisé :
  L'apprentissage automatique supervisé consiste à prédire une variable cible appelée target y à partir d'un vecteur X de variables explicatives observées appelées features, tout cela à l'aide d'un modèle f tel que $f(X) = y$. C'est dans cette famille que l'on retrouve les modèles de classification et de régression. (Exemples : Modèles de *Boosting*, réseaux de neurones, arbres de décision, etc…)

- Les algorithmes d'apprentissage non supervisé :
  L'apprentissage automatique non supervisé permet d'identifier des patterns communs ou d'effectuer des regroupements de données sans nécessiter d'intervention humaine. Il utilise des données « non étiquetées ». Il est adapté à des problèmes de *clustering*. (Exemples : modèles *k-means*, classification hiérarchique, et analyse des composantes principales (ACP), etc…)

Notre objectif étant d'estimer la durée jusqu'au rachat sur la base de nos observations, nous sommes dans une problématique de régression. On rappelle également que cette durée est calculée par la différence entre la date de souscription et la date de rachat total, celle-ci étant exprimée en mois.

Plusieurs modèles de *Machine Learning* sont capables de résoudre des problématiques de régression, notre choix s'est porté sur l'algorithme *XGBoost*, une implémentation dérivée du *Gradient Boosting Machine*. Ce choix a été motivé sur des critères de performance, de rapidité d'exécution et d'adaptabilité à notre problématique (le modèle *XGBoost* étant adaptable à des données censurées).

3.1.1. Cadre théorique

L'algorithme *XGBoost* (pour *eXtreme Gradient Boosting*) est un algorithme qui repose sur l'approche *Gradient Boosting Machine*, développé par Friedman en 2001. Il s'agit d'une méthode qui permet l'agrégation de classifieurs faibles (des arbres de décision) en utilisant l'algorithme de descente de gradient pour l'optimisation de la fonction de perte. Le principe est de combiner les résultats d'un ensemble de modèles plus simples et plus faibles pour fournir une meilleure prédiction.

Nous expliquons ci-dessous les différentes mécaniques constituant le modèle *XGBoost*.

3.1.1.1. Arbres de décision

Les arbres de décision de type *CART* (acronyme de *Classification and Regression Trees*) sont des méthodes d'apprentissage supervisé permettant de répondre à des problématiques de classification ou de régression. Ces modèles ont l'avantage de disposer d'un pouvoir explicatif simple. Les prédictions obtenues sont présentées sous la forme d'un graphique facilement interprétable. Ces arbres constituent la base de l'algorithme *XGBoost*.

La construction d'un arbre est fondée sur une séquence récursive de règles de décisions :

- Le nœud initial se trouve à la racine de l'arbre et comprend l'ensemble de l'échantillon.
- Chaque nœud est formé par le choix d'une variable explicative et d'une valeur seuil permettant de partitionner l'échantillon en deux sous-groupes, la procédure est ensuite itérative.

L'algorithme *CART* nécessite toutefois de définir un critère de sélection de la meilleure division de l'échantillon possible à partir de la sélection d'une variable explicative, mais également de trouver une règle permettant d'arrêter le partitionnement et la création de nouveaux nœuds, le nœud terminal se transformant en feuille.

Pour déterminer les prédictions finales de chaque feuille sur les nœuds terminaux de l'arbre, on dispose de 2 cas selon la problématique dans laquelle on se trouve :

- Classification : la classe prédite correspond à la classe majoritaire de la feuille.
- Régression : la valeur prédite correspond à la moyenne de la variable cible calculée à partir des observations contenues dans la feuille.

Prenons l'exemple d'arbre de régression suivant. Les conditions « X[0] < 5 ? », « X[3] < 10 ? » et « X[2] < -2 ? » permettent d'effectuer 2 divisions successives et ainsi obtenir 4 feuilles (chaque nœud parent engendrant deux fils). Ces 4 feuilles nous permettent d'effectuer nos prédictions finales.

Illustration 13 : Exemple d'arbre de régression

Pour chaque partitionnement de l'arbre, l'algorithme *CART* va chercher à effectuer ses séparations de sorte à former les sous-groupes les plus homogènes possibles.

Par rapport à des modèles de régression linéaire, les algorithmes *CART* ont plusieurs avantages :

- L'approche par arbre est plus simple et directe.
- La structure reliant la variable cible aux variable explicatives ne nécessite pas d'être linéaire.
- Les dépendances entre les variables explicatives ne posent pas de problème.

3.1.1.2. Gradient Boosting Machine & XGBoost

Comme introduit précédemment, le modèle *XGBoost* fonctionne par l'agrégation des résultats de plusieurs *CART*, il repose sur un système additif.

Il existe deux principales méthodes d'apprentissage ensembliste :

- D'une part, le *bagging* : Chaque arbre crée est considéré comme un « *weak learner* » (un classifieur dit « faible ») autrement dit, il s'agit d'un arbre ayant une faible performance prédictive. Cependant, l'idée est de sommer les prédictions de tous ces arbres pour obtenir une prédiction fiable.

- D'autre part, le *boosting*, utilisé par l'algorithme *XGBoost* : A chaque itération de l'algorithme (il faut entendre par là chaque création d'un nouvel arbre *CART*), le modèle ajouté a pour but de corriger les erreurs commises par les arbres précédents.

Le Gradient Boosting est donc une évolution de l'algorithme de *boosting*. Il consiste en la création de classifieurs faibles, de manière itérative et corrigeant les erreurs des classifieurs précédents. L'idée est de créer un premier classifieur très basique, l'algorithme calcule l'écart entre les prédictions et la réalité, c'est que l'on appelle les résidus. Dans son principe de fonctionnement, le classificateur suivant va être entrainé pour améliorer les prédictions du modèle. Chaque classifieur est ensuite pondéré en fonction de la performance de la prédiction. Cette pondération est calculée par l'utilisation de la descente de gradient. A mesure que l'algorithme créée de nouveaux arbres qui minimisent les résidus, le pas devient de plus en plus précis pour permettre aux prédictions de se rapprocher de la réalité.

L'algorithme XGBoost est une version améliorée du *Gradient Boosting* vu auparavant. Il repose sur le même principe mais est différent dans la conception des classifieurs faibles utilisés. Dans cet algorithme, les arbres décisionnels qui ne sont pas assez bons sont « élagués », c'est-à-dire que certaines de leurs branches sont supprimées, voire l'arbre entier, jusqu'à obtention d'un classifieur performant.

Ainsi construit, l'algorithme *XGBoost* permet de répondre à des problématiques de régression en construisant des modèles sur un ensemble complexe de *features*.

3.2. Approche de notre modélisation

3.2.1. Echantillonnage

Dans un modèle de *Machine Learning*, le but est de construire un modèle de prédiction. On peut représenter cela comme une fonction $f(.)$ qui prend un echantillon de données en entrée et prédit une valeur en sortie que l'on notera $\hat{y}$. Dans le cas d'un algorithme d'apprentissage supervisé où l'on souhaite prédire une valeur, l'ensemble de données est généralement subdivisé en 3 parties, on distingue :

- Un échantillon d'apprentissage du modèle, l'échantillon principal. L'algorithme effectue son apprentissage sur ces données, il sert à ajuster le modèle.

- Un échantillon de validation qui permet d'éviter le phénomène de surapprentissage. Ce phénomène se présente lorsque le modèle apprend les bruits de l'échantillon d'apprentissage. Les prédictions de ce modèle seront donc nettement moins précises sur un nouveau jeu de données, puisque le modèle tentera de répliquer ce bruit.

- L'échantillon de test est lui utilisé pour évaluer le modèle conçu. Il permet d'estimer la qualité réelle de la capacité prédictive du modèle. Ces données étant indépendantes de l'échantillon d'apprentissage, elles n'ont pas été utilisées pour la calibration du modèle.

Illustration 14 : Schéma d'échantillonnage du jeu de données

Le principe est le suivant :

- On divise notre base en trois échantillons : *train* (apprentissage), *eval* (validation) et *test*.
- Le modèle est entrainé sur les données de la base d'apprentissage et on observe les prédictions sur la base de validation à chaque itération.
- Nous utiliserons dans nos développements la méthode dite *d'early stopping.* Elle consiste à arrêter l'apprentissage du modèle lorsque qu'il n'observe plus de diminution de l'erreur sur l'échantillon de validation sur ses n dernières itérations. Autrement nous pourrions également fixer un nombre d'itération prédéfini et laisser le modèle aller au bout de celles-ci. Cependant, *l'early stopping* est plus efficient et permet au modèle de limiter le phénomène de surapprentissage.
- Le modèle ainsi construit, nous évaluons finalement sa qualité en comparant les prédictions sur la base de test à leur valeur réelle.

Pour notre problématique nous souhaitions dans une première approche nous placer dans un cadre hypothétique où nous connaissions les rachats d'année en année, c'est-à-dire que dans un premier temps nous ne prenons pas en compte les données censurées (nous utiliserons la variable cible simplifiée explicitée précédemment dans la section 2.1.3). Il s'agit là d'évaluer la capacité de modélisation d'une année sur l'autre.

L'idée est de récupérer les observations de rachat des années précédentes, pour estimer les durées jusqu'au rachat des années futures. Pour contenir le biais lié à l'évolution de l'ancienneté des contrats, nous avons choisi de prendre une période de 15 ans entre la date de souscription et le rachat.

Ainsi, notre base d'entrainement et d'évaluation est constituée sur une répartition 85%/15% des contrats disposant des caractéristiques suivantes :

- Les contrats rachetés en 2017 et souscrits à partir de 2002
- Les contrats rachetés en 2018 et souscrits à partir de 2003

Notre base de test elle, est constituée des contrats rachetés en 2019 et souscrits à partir de 2004.

Ce découpage non aléatoire entre l'échantillon d'apprentissage et l'échantillon de test permet de prendre en compte une modélisation rétrospective des durées de rachats.

3.2.2. Choix des hyperparamètres

L'algorithme *XGBoost* étant relativement long et complexe du fait de la présence d'un grand nombre de hyperparamètres. Ces hyperparamètres représentent les propriétés générales de l'algorithme et ne sont pas modifiées pendant la phase d'apprentissage. Ils doivent être défini en amont du calibrage de l'algorithme.

Dans le cadre de notre étude, nous avons décidé d'optimiser les hyperparamètres suivants exerçant une influence forte sur la capacité d'entrainement du modèle :

- max_depth : Il s'agit du paramètre gérant la profondeur maximale des arbres générés. L'augmentation excessive de cette valeur rend le modèle plus complexe et accroît le risque de sur-apprentissage.

- colsample_bytree : Il s'agit du paramètre déterminant le pourcentage de variables explicatives utilisées à chaque itération.

- gamma : C'est le paramètre de régularisation, il s'agit de la réduction minimale de la fonction de perte requise afin d'effectuer un nouveau partitionnement de nœud. Plus gamma est grand, plus l'algorithme sera conservateur.

- **min_child_weight** : C'est un paramètre de contrôle de la complexité du modèle. Si lors de l'étape du partitionnement d'un nœud de l'arbre, les poids associés à l'une des feuilles se retrouvent être inférieurs à min_child_weigth, alors l'algorithme stoppera le partitionnement de l'arbre. Plus ce paramètre est grand, plus l'algorithme sera conservateur.

L'optimisation des paramètres précédents a pour but d'éviter le phénomène de surapprentissage, commun dans les problématiques de *Machine Learning*.
Afin d'optimiser les paramètres précédents, nous avons utilisé la **méthode de validation croisée** sur notre échantillon d'apprentissage. Il s'agit d'une méthode d'estimation de fiabilité d'un modèle basé sur une technique d'échantillonnage. On divise notre base d'apprentissage en k sous-groupes (dans notre cas nous avons choisi k=10). On sélectionne ensuite un des k sous-groupes comme base de test pendant que les $k-1$ autres sous-groupes constituent la base d'apprentissage. Cette opération est répétée de façon à ce que tous les sous-groupes aient été la base de test retenue. La fonction *cv(.)* du package *XGBoost* permet d'évaluer un modèle en utilisant la méthode de validation croisée et renvoie le score moyen des modèles formés sur chaque sous-groupe. Ce score moyen est calculé à l'aide d'une fonction de coût.

Dans notre cas, la fonction de coût choisie à minimiser est **l'erreur quadratique**.

Nous définissons dans un premier temps les domaines de distribution de nos hyperparamètres.

Hyperparamètre	Distribution
max_depth	Entier compris dans [5 , 20]
colsample_bytree	Réel compris dans [0.5 , 1.0]
gamma	Réel compris dans [0 , 0.5]
min_child_weight	Entier compris dans [2 , 6]

Notre problématique est la suivante : trouver les valeurs des hyperparamètres qui minimisent la fonction de coût du modèle *XGBoost* calculé par la méthode de validation croisée.

Une possibilité serait d'essayer chaque combinaison d'hyperparamètres et de conserver ceux atteignant le minima global de cette fonction de coût. Cependant, bien que de tels algorithmes soient implémentés (ex : *GridSearch*), cette solution serait couteuse en temps de calcul. La méthode d'optimisation bayésienne permet d'éviter cela en trouvant le minimum global en un nombre d'étapes réduit.

Cette méthode consiste en la construction d'une distribution a posteriori qui représente le mieux la fonction que l'on cherche à optimiser. La méthode d'optimisation teste plusieurs valeurs d'hyperparamètres dans leur espace de définition (tableau ci-dessus). À mesure que le nombre d'observation augmente, la distribution a posteriori se rapproche de la vraie fonction de coût. L'algorithme cible son exploration dans les zones où se trouve le potentiel minimum global.

Pour notre cas d'étude, nous avons trouvé que les hyperparamètres suivant étaient optimaux :

- max_depth = 14
- colsample_bytree = 0.541
- gamma = 0.0697
- min_child_weight = 3

3.2.3. Premier modèle : Utilisation de la Root Mean Squared Error

3.2.3.1. Présentation du cadre

Pour pouvoir mesurer la performance d'un algorithme et donc du modèle utilisé, il faut pouvoir calculer l'erreur des prédictions par rapport aux valeurs réelles.

Ainsi, pour notre premier modèle, nous utiliseront la métrique *RMSE* (pour *Root Mean Squared Error*) qui est la racine carrée de l'erreur quadratique moyenne. Elle calcule la moyenne des écarts entre les prédictions du modèle et les valeurs observées, le but du modèle étant de minimiser cette quantité :

$$loss_{RMSE} = \sqrt{\sum_{i=1}^{n} \frac{(\hat{y}_i - y_i)^2}{n}}$$

Nous construisons notre modèle avec cette métrique car elle est sans-biais. Ce qui nous permettra de prendre ce modèle comme référence pour des comparaison avec des modèles biaisés. En revanche, dans le cadre de comparaison avec d'autres estimateurs sans biais, le modèle disposant de la variance la plus faible est considéré comme étant le plus efficace.

Pour rappel, dans le cadre de cette modélisation nos 3 échantillons sont les suivants :

- La base d'entrainement et base de validation sont composés des contrats rachetés en 2017 souscrits à partir de 2002 et des contrats rachetés en 2018 souscrits à partir de 2003.

- La base de test est constituée des contrats rachetés en 2019, souscrits à partir de 2004.

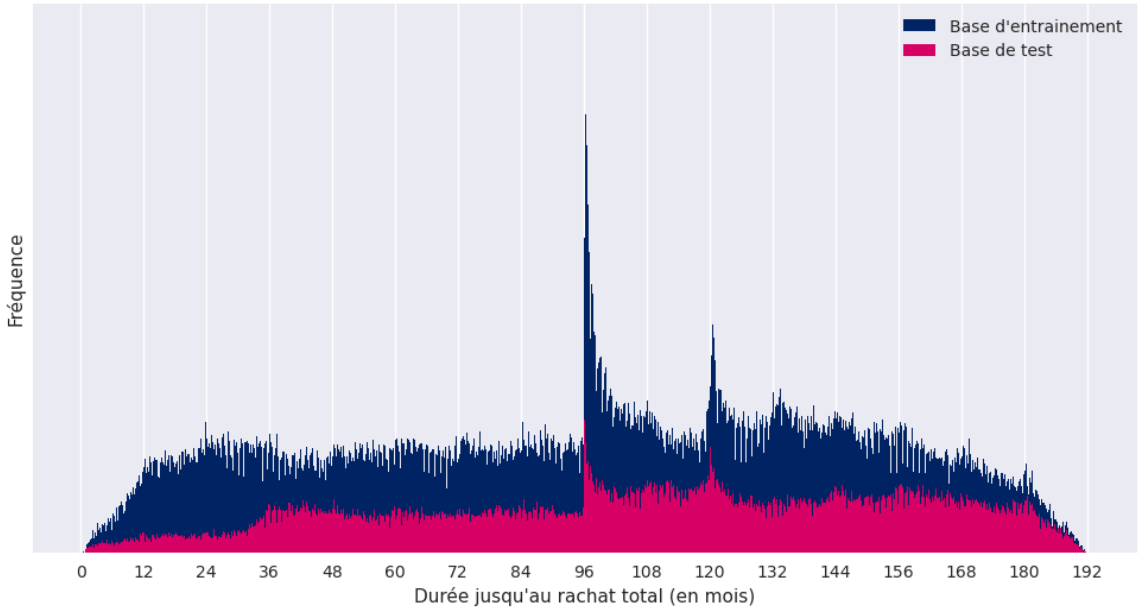

Illustration 15 : Histogramme superposé de la variable cible entre la base train et la base test.

Le graphique précédent représente la distribution superposée de la durée jusqu'au rachat dans les échantillons d'apprentissage et de test.

On observe une distribution en proportions similaires de la variable à prédire pour les durées supérieures à 3 ans. On dénote une plus faible proportion des durées inférieures à 3 ans pour la base de test.

3.2.3.2. Modélisation et résultats

À la suite de l'entrainement d'un modèle *XGBoost* sur ces données d'apprentissage à l'aide des hyperparamètres définis dans la section précédente, nous obtenons le graphique suivant qui nous infome de l'importance des *features* (variables explicatives) ainsi que de leur apport dans le calcul de la prédiction.

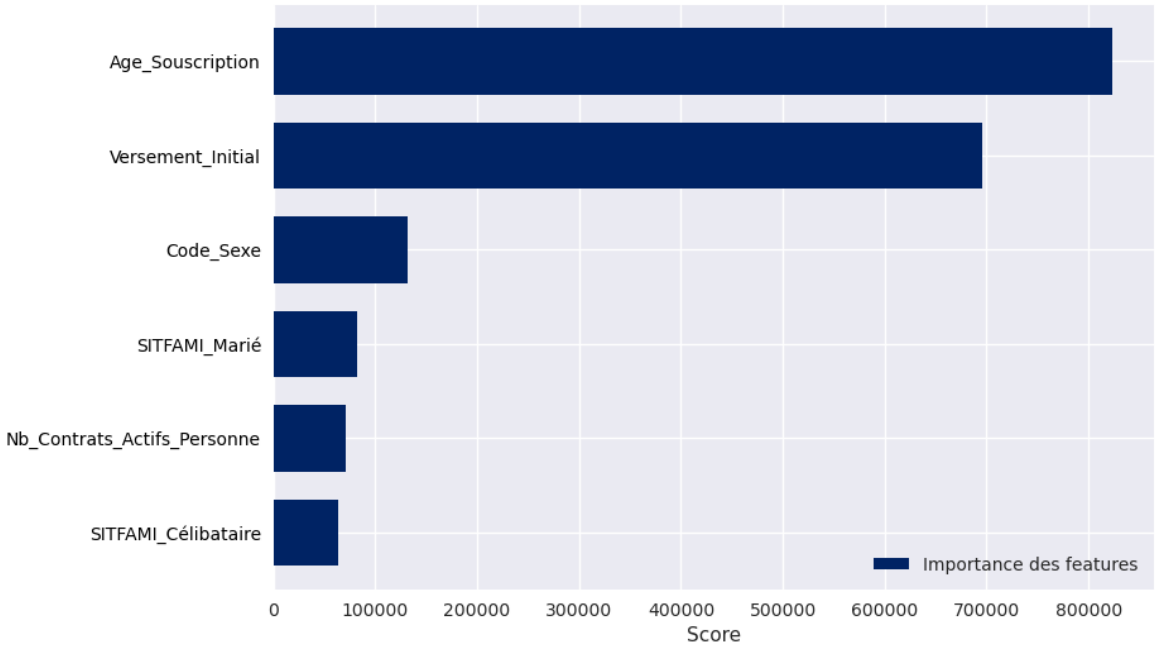

Illustration 16 : Importance des *features* du modèle « RMSE »

L'importance des *features* peut être définie de plusieurs méthodes différentes, dans notre cas il s'agit du nombre de fois où la variable explicative est utilisée pour partitionner les données à l'intérieur des arbres de régression crées par le modèle. A noter également que les résultats auraient pu être différents selon la méthode utilisée. Par exemple une autre méthode consiste à ordonner l'importance des variables selon le gain de score moyen des partitionnements impliquant ladite variable.

On remarque que les deux variables les plus discriminantes du modèle pour la prédiction de la durée jusqu'au rachat sont l'âge de l'assuré à la souscription et le montant de versement initial. Ainsi, sur les données disponibles à la souscription du contrat, les variables qui jouent le plus sur la durée du rachat sont celles qui permettent d'identifier le profil et la catégorie sociale du client, de par son âge et sa capacité d'épargne.

Nous cherchons maintenant à évaluer le potentiel prédictif du modèle. Les deux mesures d'erreur suivantes que nous introduisons ne sont pas celles utilisées pour optimiser le modèle, elles servent uniquement à mesurer la précision des prédictions $\hat{y}$ par rapport aux données réelles y sur l'échantillon de test. Il s'agit des deux mesures classiques utilisées dans les problématiques de régression.

- L'erreur moyenne absolue, ou MAE (pour *Mean Absolute Error*), il s'agit de la moyenne arithmétique de la valeur absolue des écarts. C'est un estimateur sans biais qui compare la magnitude moyenne des erreurs, toutes les erreurs ont le même poids.

$$MAE = \frac{\sum_{i=1}^{n} |\hat{y}_i - y_i|}{n}$$

- L'erreur quadratique, ou MSE (pour *Mean Square Error*), il s'agit de la moyenne des carrés des erreurs. Cet estimateur est également sans biais, cependant les résidus n'ont ici pas le même poids, c'est une représentation de la variance des résidus.

$$MSE = \frac{\sum_{i=1}^{n} (\hat{y}_i - y_i)^2}{n}$$

Nous prendrons comme base de comparaison la durée moyenne jusqu'au rachat en fonction des groupes de risque homogènes.

Nous obtenons les valeurs suivantes dans notre échantillon de test :

- Moyenne des durées pour le GRH « Haut de Gamme » : 100.33 mois
- Moyenne des durées pour le GRH « Grand Public Agés » : 111.39 mois
- Moyenne des durées pour le GRH « Grand Public Jeune » : 107.31 mois

	MAE Modèle	MAE Moyen	MSE Modèle	MSE Moyen
Base totale	29.1	39.2	1348.7	2176.6
GRH « Haut de Gamme »	20.3	52.6	748.8	3449.8
GRH « Grand Public Agés »	27.8	35.2	1265.0	1778.7
GRH « Grand Public Jeune »	33.5	39.5	1636.7	2230.9

Précision : Les catégories « MAE Moyen » et « MSE Moyen » sont les calculs d'erreurs entre la durée observée réelle et la durée moyenne calculée sur la base des regroupements par GRH.

Dans ce tableau, on remarque que le modèle effectue des prédictions en moyenne plus précises par rapport aux durées regroupées par GRH. Lorsqu'on se concentre sur les contrats « Haut de Gamme », on observe que les prédictions sont deux fois plus précises.

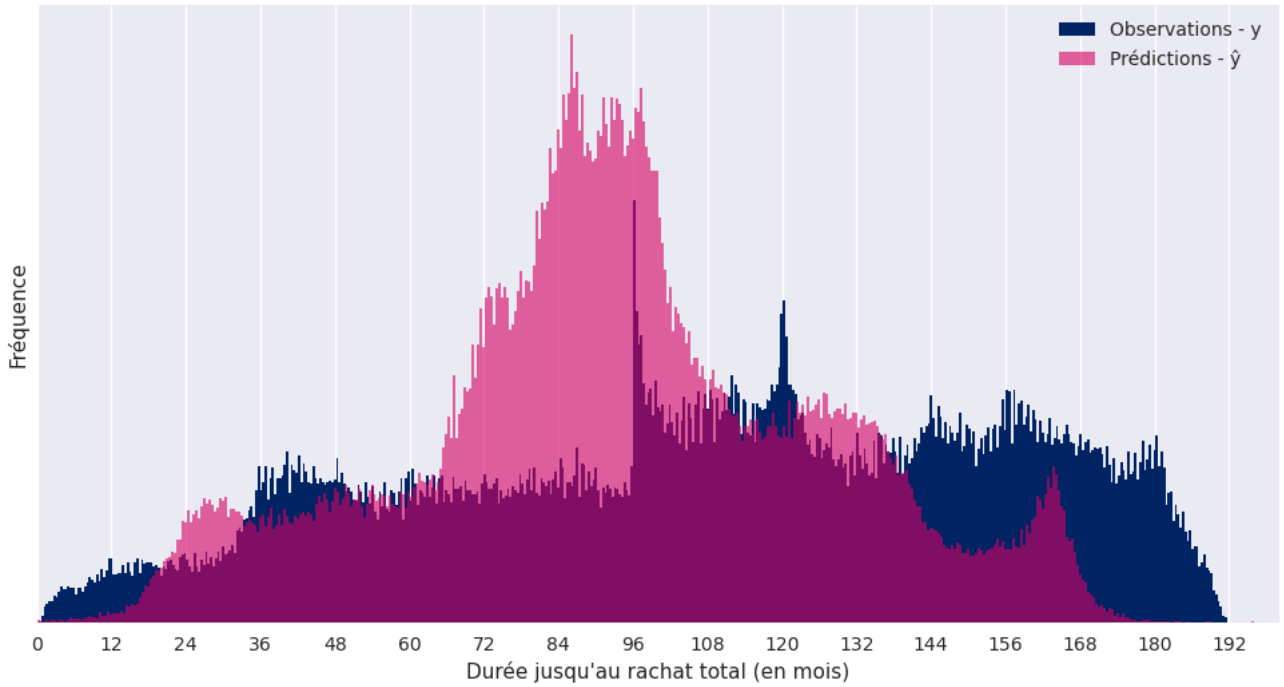

Illustration 17 : Distribution des prédictions du modèle

Ce graphique représente les distributions des prédictions (en rose) et des valeurs observées (en bleu), on remarque que le modèle capte bien la hausse du nombre de rachats vers la 8ème année (96ème mois), mais ce pic n'est pas retranscrit de manière précise dans les prédictions, on dispose à la place d'une grande proportion de valeurs prédites se situant entre le 72ème et le 108ème mois (6-9 ans).

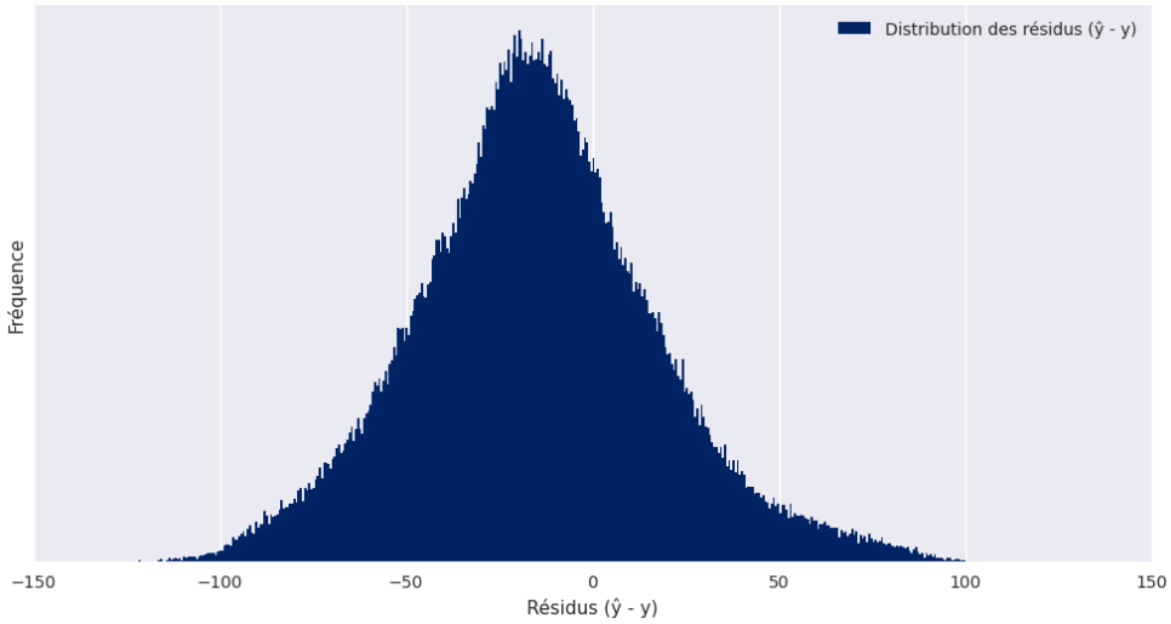

Illustration 18 : Distribution des résidus sur la base de test

Le graphique ci-dessus représente la distribution des résidus (les écarts entre la prédiction et la valeur observée) sur la base de test. Une valeur positive correspond à une sur-prédiction (la durée prédite est supérieure à celle observée réellement), à l'inverse une valeur négative représente une sous-prédiction.

On observe que notre modèle a une légère tendance à sous-prédire la durée jusqu'au rachat, cette observation est à nuancer puisque cela pourrait être dû à la grande majorité des contrats ayant été rachetés lors de la 8ème année d'ancienneté, ces cas où notre modèle prédit les durées sur une période commençant 2 ans avant le pic.

Ces premières observations nous permettent de conclure que même sans les données de la vie du contrat, avec seulement les données disponibles à la souscription, le modèle capte les tendances de rachats dans une certaine mesure, mais reste peu précis dans l'exécution et la prédiction individuelle des durées. En moyenne le modèle se trompe de 3 ans dans sa prédiction, mais il reste cependant plus précis que l'approche par regroupement de contrats dans des groupes de risque homogènes.

3.2.4. Second modèle : MSE avec fonction de perte personnalisée

3.2.4.1. Présentation du cadre

Ce second modèle implémente la notion de métrique personnalisée, qui correspond mieux à notre problématique.

D'un point de vue opérationnel, on préfère être plus prudent quant à nos prédictions, c'est-à-dire prédire une durée inférieure à la durée réelle. Comme on a pu le voir dans la section précédente avec l'utilisation de la RMSE, un bon nombre des prédictions étaient effectuées au-delà de la durée de rachat effective. Donc dans notre étude, on préférera anticiper l'événement de rachat.

L'objectif de ce modèle est donc de sous-prédire les durées. Pour faire cela sans modifier la structure des données, on peut modifier la façon dont s'entraine le modèle pour pénaliser les sur-prédictions (prédire une valeur supérieure à sa valeur réelle).

Ce type de changement est implémentable dans le modèle *XGBoost*. Il s'agit de modifier la fonction de perte en une fonction qui pénaliserait plus les valeurs positives que négatives. De plus, il est nécessaire que cette fonction soit deux-fois différentiable. *XGBoost* requiert pour l'implémentation de chaque fonction de perte, sa dérivée première et seconde. Il faut que celles-ci soient continues.

Une manière d'effectuer cela est de reprendre la fonction d'erreur quadratique moyenne, fonction deux-fois différentiable, et de lui ajouter un facteur $k > 1$ lorsque le résidu prend des valeurs positives. On rappelle que le résidu correspond à la différence entre la prédiction et sa valeur réelle.

Ainsi notre fonction de perte globale est définie comme suit :

$$loss_{custom}(\hat{y}, y) = \sqrt{\frac{1}{n}\sum_{i=1}^{n} err_i^{\,2} * 1_{\{err_i \leq 0\}} + k * err_i^{\,2} * 1_{\{err_i > 0\}}}, \qquad avec\ err_i = \hat{y}_i - y_i$$

Pour chaque écart entre la prédiction du modèle et la valeur réelle, la fonction de perte personnalisée va pénaliser les sur-prédictions. Le tableau ci-dessous compare les fonctions de pertes RMSE (utilisée pour le calibrage du modèle précédent), et la fonction de perte personnalisée que nous appellerons « *Custom Loss* ». Nous avons choisi la constante $k = 4$ pour suffisamment pénaliser la sur-prédiction sans que cela soit extrême pour que l'algorithme ne créé pas un modèle excessivement biaisé.

Nous prenons l'exemple ci-dessous pour illustrer nos propos :

Valeur réelle	Valeur prédite	RMSE	Custom Loss (k=4)
100	80	20	20
100	120	20	40

On remarque que notre fonction de perte personnalisée pénalise bien plus la sur-prédiction que la RMSE.

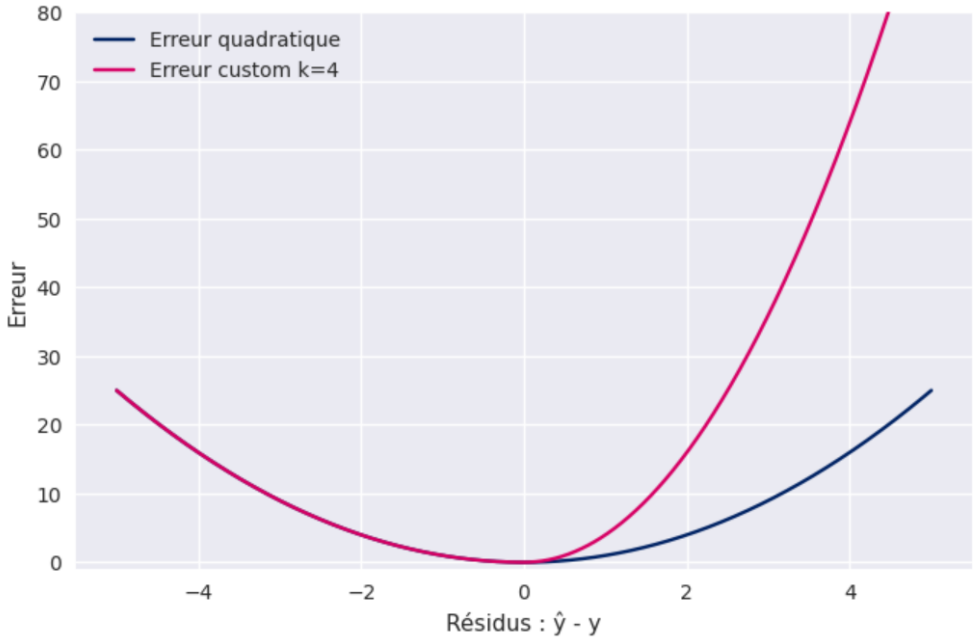

Illustration 19 : Comparaison graphique des fonctions de perte avant passage à la racine

Nous retrouvons ci-dessus la comparaison graphique des fonctions de pertes quadratiques (en bleu) et de notre métrique personnalisée (en rouge). On observe bien la pénalisation croissante de la sur-prédiction.

3.2.4.2. Modélisation et résultats

Dans le but de pouvoir comparer les comportements des deux modèles à isopérimètre donné, nous calibrons le modèle « *Custom-Loss* » avec les mêmes données et hyperparamètres que le modèle précédent.

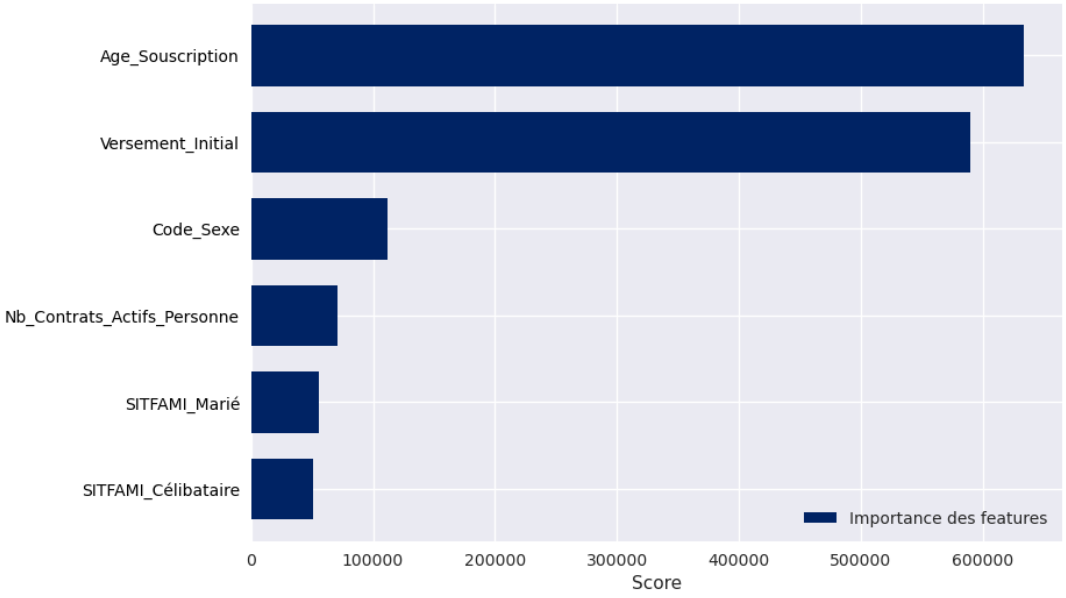

Illustration 20 : Importance des *features* du modèle « *Custom-Loss* »

Le graphique précédent nous informe de l'importance des *features* dans le modèle « *Custom-Loss* ». On remarque que les deux variables les plus explicatives n'ont pas changé, seulement l'ordre d'importance des variables « secondaires » l'est.

Afin de mesurer le potentiel prédictif du modèle nous utiliserons les mêmes métriques d'erreur que dans la section précédente, à savoir la MAE et MSE.

- Erreur moyenne absolue (MAE)

	Modèle « Custom-Loss »	Modèle « RMSE »	Moyenne GRH
Base totale	36.8	29.1	39.2
GRH « Haut de Gamme »	25.5	20.3	52.6
GRH « Grand Public Agés »	35.6	27.8	35.2
GRH « Grand Public Jeune »	41.8	33.5	39.5

- Erreur quadratique (MSE)

	Modèle « Custom-Loss »	Modèle « RMSE »	Moyenne GRH
Base totale	2099.9	1348.7	2176.6
GRH « Haut de Gamme »	1174.2	748.8	3449.8
GRH « Grand Public Agés »	1979.7	1265.0	1778.7
GRH « Grand Public Jeune »	2533.9	1636.7	2230.9

Les mesures d'erreur précédentes nous montrent que de manière générale, le modèle dispose d'un pouvoir prédictif similaire à l'utilisation des durées moyennisées par GRH, qui par définition sur-prédisent les durées sur 50% des cas. Ce qui ne devrait pas être le cas pour les prédictions de ce modèle.

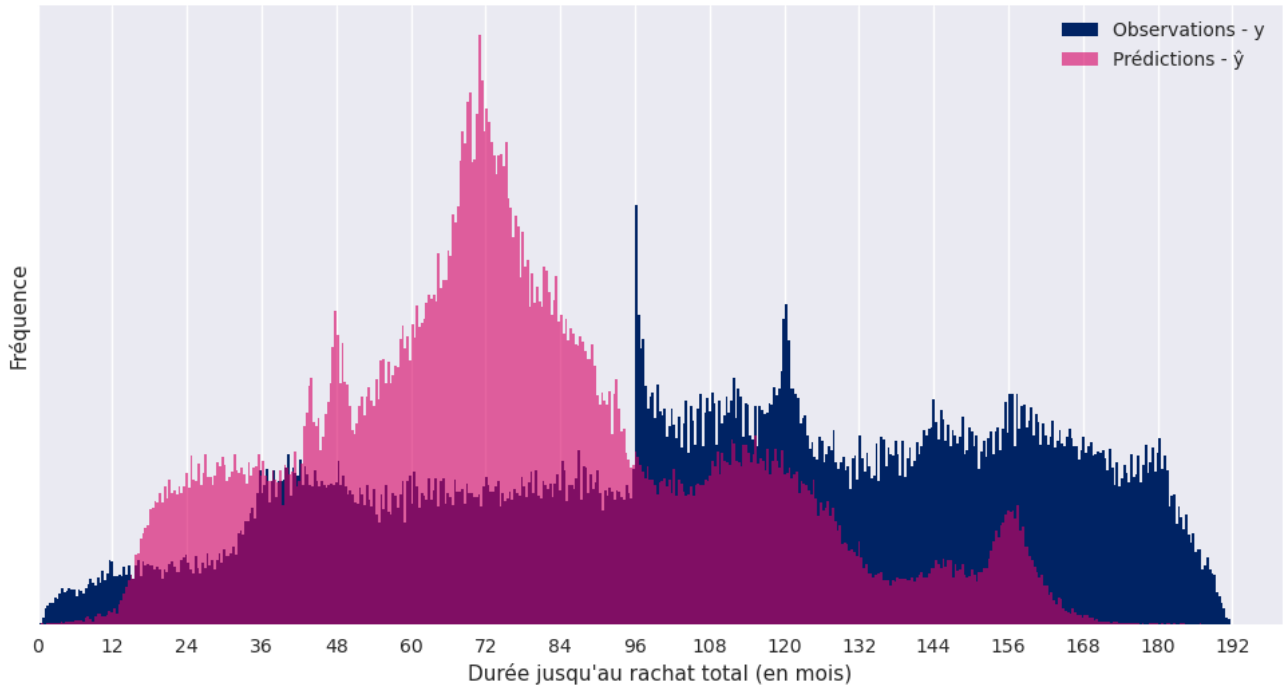

Illustration 21 : Distribution des prédictions du modèle « *Custom-Loss* »

Le graphique ci-dessus représente les distributions des prédictions (en rose) et des valeurs réelles (en bleu), on remarque le biais des prédictions. Le modèle capte le pic de rachats de la 8ème année, mais la tendance sous-prédictive du modèle réintègre la hausse subite des rachats 2 ans avant le pic de rachat effectif. Cependant, ce pic dans les prédictions est plus concentré que celui du premier modèle.

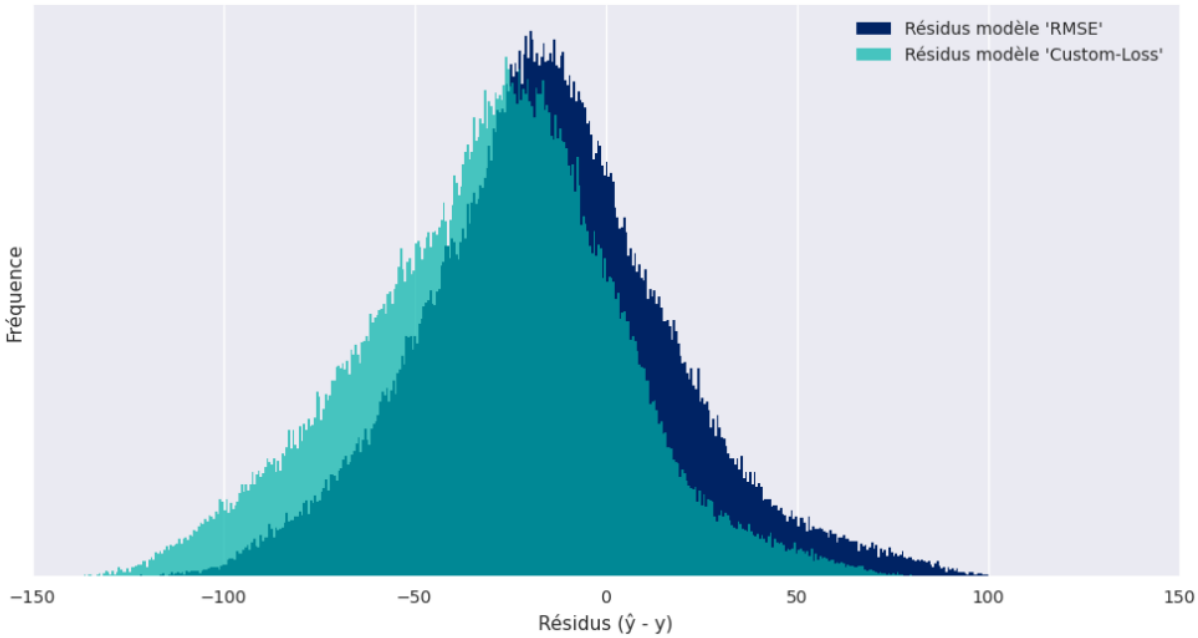

Illustration 22 : Comparaison des résidus des deux modèles

Le graphique ci-dessus compare la distribution des résidus pour le premier et le second modèle sur la base de test.

On remarque que la distribution s'est translatée vers les valeurs négatives, avec une descente dans la distribution des sur-prédictions légèrement plus prononcée que pour le premier modèle.

De manière générale, l'utilisation de cette fonction de perte personnalisée pénalisant les sur-prédictions à introduit un biais important dans les prédictions. Cela a cependant permis de réduire de moitié le nombre de rachats en sur-prédiction. Dans le cas du premier modèle, on disposait de 30% de sur-prédiction, grâce au deuxième modèle ce chiffre baisse de moitié et atteint 16%.

3.2.5. Troisième modèle : Modélisation avec fonction de survie

Depuis peu, certaines méthodes de *Machine Learning* se sont adaptées pour rendre possible la modélisation de durée, c'est le cas du *XGBoost* par le biais d'une implémentation d'un modèle d'analyse de survie : AFT (pour *Accelerated Failure Time*) et de la librairie *xgbse* (*XGBoost Survival Embedding*) de Python.

Ces deux implémentations permettent d'accéder à une fonctionnalité importante : la prise en compte des phénomènes de censures, phénomène fréquemment rencontré dans les problématiques d'étude de durées.

En effet, dans les deux modèles précédents nous avions choisi une approche hypothétique pour mesurer la capacité de généralisation d'une modélisation dans les cas où le sinistre est connu. Or, si on sort de ce cadre hypothétique on remarque que les données censurées représentent la grande majorité de nos observations. Pour répondre à la problématique initiale sans induire de biais nous devons élargir notre étude aux données censurées.

3.2.5.1. Introduction aux modèles adaptés aux problématiques de survie

L'implémentation *XGBoost* native fournit deux méthodes d'analyse de survie : *Cox* (Chen et Guestrin, 2016) et *AFT* (Barnwall, Cho et Hocking, 2020). Lorsqu'il s'agit de classer les individus par risque (de les discriminer), ces deux modèles affichent des performances compétitives (mesurées par l'indice de concordance C, l'équivalent de l'AUC pour la survie) tout en restant efficaces en termes de calcul.

Cependant, on observe des lacunes en ce qui concerne certaines propriétés statistiques, principalement :

- La prédiction de courbes de survie plutôt que des estimations ponctuelles
- L'estimation d'intervalles de confiance.
- Des temps de survies attendus calibrés (et non biaisés)

Bien que nécessitant une extension pour améliorer ces propriétés statistiques, *XGBoost* n'en reste pas moins un modèle puissant. Les résultats de l'indice de concordance montrent que le modèle a de grandes performances de discrimination, tout en étant compétitif avec l'état de l'art. Il suffit donc de trouver le moyen de l'adapter à l'utilisation visée.

C'est dans ce contexte qu'une version du modèle *XGBoost* a été proposée dans le cadre de l'analyse de survie (*XGBoost Survival Embedding*) (Viera et al., 2021).

3.2.5.2. Utilisation de XGBoost comme encodeur

Bien qu'utilisées pour des tâches de prédiction, les méthodes de *Gradient Boosting* peuvent également être utilisées comme des *transformer* de données d'entrée (agissant comme un encodeur). Les arbres effectuent des divisions permettant de discriminer la valeur cible, en encodant dans leur structure les informations les plus pertinentes pour la tâche visée. En

particulier les nœuds terminaux (feuilles) de chaque arbre de l'ensemble qui définissent une transformation de *features* (appelé *embedding*) des données d'entrées.

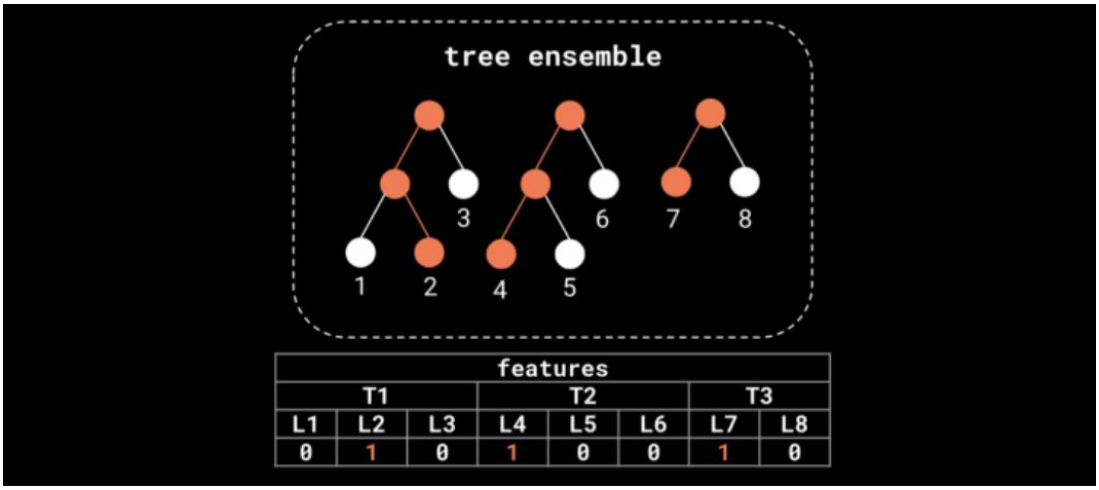

Illustration 23 : Utilisation de *XGBoost* comme encodeur

La figure ci-dessus illustre un cas d'usage avec trois arbres. On peut extraire des caractéristiques d'un modèle XGBoost, en transformant l'espace de caractéristiques original en un *embedding* « d'occurrence de feuilles ». Les nœuds orange représentent le chemin d'un seul échantillon dans l'ensemble.

Ce type d'*embedding* d'ensemble d'arbre possède des propriétés caractéristiques pratiques :

- sparcité et haute dimensionnalité : les arbres traitent la non-linéarité et transforment les caractéristiques d'origine en un *embedding* à haute dimensionnalité, permettant à des modèles linéaires de proposer de bonne performances lorsqu'ils sont entrainés dessus. Cela permet par exemple à une régression logistique entrainée sur l'*embedding* d'avoir des performances comparables à celles de l'ensemble réel avec l'avantage supplémentaire de la calibration de probabilités. (He et al., 2014), (Marmerola, 2018)
- supervision : les arbres fonctionnent également comme un filtre de bruit, effectuant des divisions uniquement sur les variables apportant un signal. De fait, l'*embedding* a une

dimension intrinsèque plus faible que les données d'entrée. Cela atténue la contrainte de la dimensionnalité et permet à un modèle K-plus proches voisins entrainé sur l'*embedding* (en utilisant la distance de Hamming) d'avoir des performances comparables à l'ensemble réel, avec la flexibilité supplémentaire de pouvoir appliquer cette fonction un ensemble des voisins pour obtenir des prédictions. Cette fonction peut être, par exemple, un estimateur de survie sans biais tel que l'estimateur de Kaplan-Meier. (Marmerola, 2018)

La distance de Hamming utilisée ici est une distance qui permet de comparer deux séquences de symboles de même longueur (ici une séquence binaire de 0 et 1), elle compte le nombre d'éléments différents entre la première et la deuxième chaîne.

Le package *xgbse* tire avantage de ces différentes propriétés, comme nous l'expliquerons plus en détail par la suite.

3.2.5.3. Présentation de XGBoost Survival Embeddings (xgbse)

Le package *xgbse* dispose de plusieurs modèles, nous détaillerons ceux utilisés dans le cadre de notre étude dans cette section.

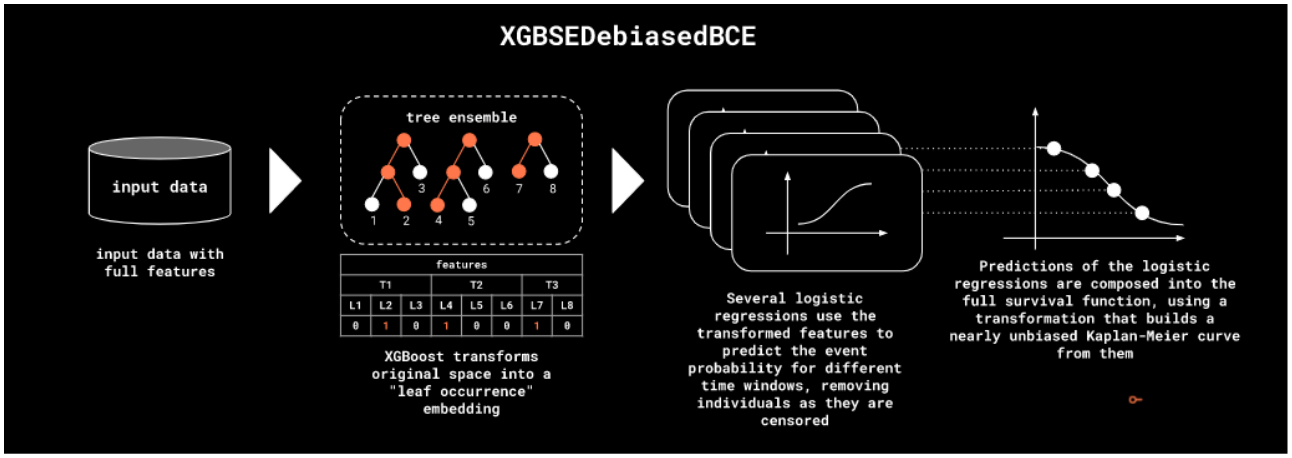

Illustration 24 : Schéma de fonctionnement du modèle *XGBSEDebiasedBCE*

Ce modèle consiste en l'entrainement de multiples régression logistiques se basant sur l'*embedding* produit par le modèle *XGBoost*, chaque régression prédisant la survie à différentes fenêtres temporelles discrètes définies par l'utilisateur. Les classificateurs éliminent les individus au fur et à mesure qu'ils sont censurés, avec des cibles qui sont des indicateurs de survie à chaque fenêtre temporelle.

L'approche naïve tend à donner des courbes de survie biaisées à cause de l'élimination des individus censurés. Il est nécessaire de procéder à quelques adaptations afin que les régressions logistiques estiment les probabilités ponctuelles dans la formule de Kaplan-Meier, puis utilisent l'estimateur KM pour obtenir des courbes de survie presque sans biais.

De cette manière, on peut obtenir des courbes de survie complètes à partir d'un modèle *XGBoost*, ainsi que des intervalles de confiance à l'aide de quelques adaptations (comme la nécessité d'effectuer quelques tours de *bootstrap*).

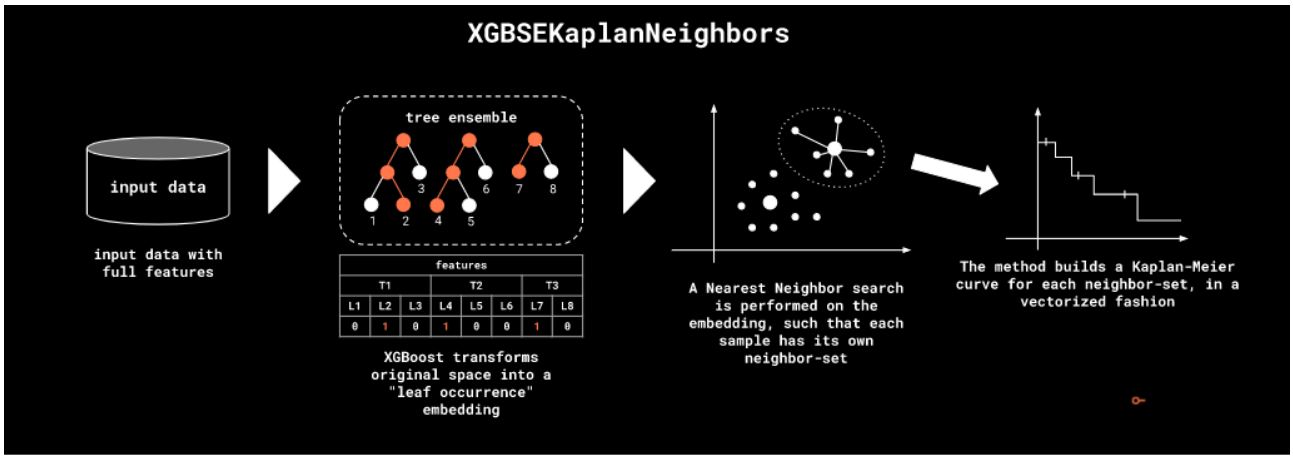

Illustration 25 : Schéma de fonctionnement du modèle *XGBSEKaplanNeighbours*

Quand bien même l'*embedding* produit par le modèle XGBoost est *sparse* et de haute dimension, sa dimension intrinsèque devrait en fait être inférieure aux données d'entrée. Cela permet de « convertir » le modèle *XGBoost* en un modèle des k plus proches voisins, la distance de

Hamming est utilisée pour définir les éléments similaires. Par la suite, sur chaque ensemble de voisins, on peut obtenir une estimation des courbes de survies grâce à l'utilisation de l'estimateur de Kaplan-Meier.

Il faut cependant garder à l'esprit que cette méthode s'avère très couteuse à l'échelle de milliers d'échantillons (comme c'est le cas dans nos données), à cause de la recherche des k-plus proches voisins. Cela affecte aussi bien l'entrainement que la prédiction (construction de l'index de recherche et de la recherche effective). Le modèle suivant tend à corriger ce problème.

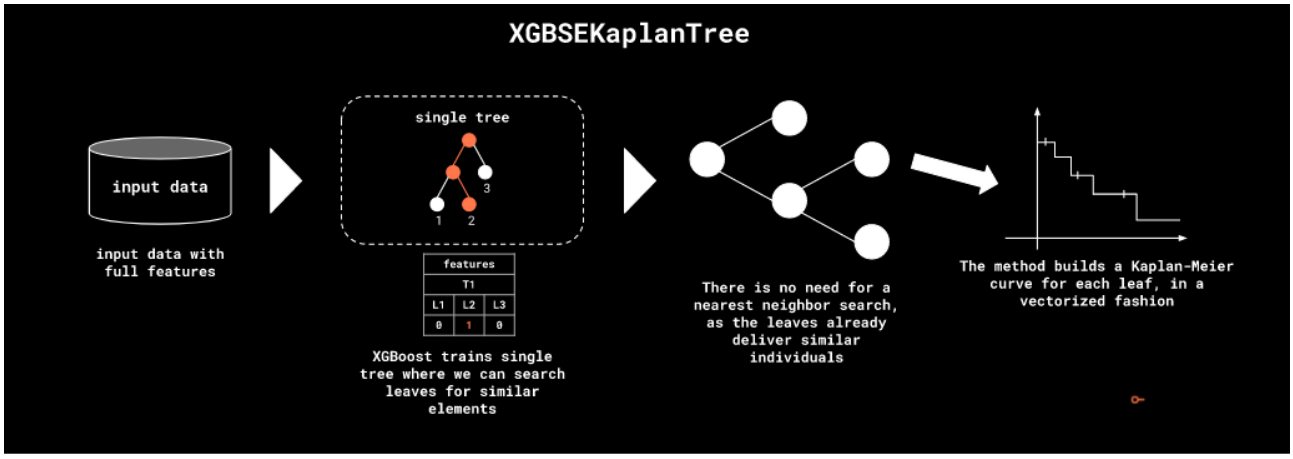

Illustration 26 : Schéma de fonctionnement du modèle *XGBSEKaplanTree*

Afin de simplifier le modèle *XGBSEKaplanNeighbours*, une implémentation a été fournie pour la construction d'un seul arbre. Au lieu d'effectuer des recherches couteuses sur les voisins les plus proches comme c'est le cas précédemment, on ajuste ici un seul arbre via *XGBoost* et on calcule les courbes de survie grâce à l'estimateur de Kaplan-Meier sur chacune de ses feuilles.

Il s'agit là de l'implémentation la plus efficace, en capacité d'évaluer aisément un échantillon de millions de données. Lors de l'entrainement, l'arbre est construit et toutes les courbes de survies sont précalculées. Au moment d'effectuer une prédiction, une requête permet d'obtenir l'estimation du modèle.

Cependant, étant donné que l'on ajuste un seul arbre, le pouvoir prédictif du modèle peut s'avérer moins bon. Dans l'objectif de limiter cet effet, une fonctionnalité (*XGBSEBoostrapEstimator*) a été développée. Il s'agit là d'une abstraction *bootstrap* permettant de construire une forêt de modèle *XGBSEKaplanTree* pour améliorer la précision et limiter la variance.

3.2.5.4. Métriques de performance

Avec l'utilisation de données censurées et l'utilisation du package *xgbse* renvoyant des probabilités de survie dans un fenêtre temporelle, l'utilisation de la MAE et MSE comme ce fut le cas avec les modèles précédents s'avère complètement inadaptée.

Ainsi, pour pouvoir évaluer la fiabilité des prédictions de nos modèles de survie nous utiliserons deux métriques implémentées dans le package *xgbse* : l'index de concordance et le score de Brier.

- Concordance Index

L'indice de concordance (ou *C-index*), est une métrique utilisée en analyse de survie qui permet de comparer la capacité prédictive des modèles. C'est une généralisation de l'aire sous la courbe ROC qui peut prendre en compte des données censurées, comme c'est le cas ici.

Il représente l'évaluation globale du pouvoir discriminant du modèle, à savoir la capacité à fournir correctement une classification des durées de survie en fonction des scores de risques individuels.

En termes d'interprétation, un indice proche de 1 correspond à la meilleure prédiction du modèle, un indice proche de 0 à une prédiction aléatoire.

- Score de Brier

Le score de Brier est utilisé pour évaluer la précision d'une prédiction d'une fonction de survie à un instant donné t. Il représente l'erreur quadratique moyenne entre le statut de survie observé et la probabilité de survie prédite et est toujours comprise entre 0 et 1, 0 étant la meilleure valeur possible. Il est calculé avec l'aide de la formule suivante :

$$BS(t) = \frac{1}{N}\sum_{i=1}^{N}(1_{T_i > t} - \hat{S}(t, x_i))^2,$$

avec N correspondant à la taille de l'échantillon, T_i la durée de survie de l'observation $i \in [1, N]$ et $\hat{S}$ la fonction de survie prédite. En termes d'interprétation, on considère qu'un modèle correct dispose d'un score de Brier inférieur à 0.25.

3.2.5.5. Modélisation

Pour la modélisation des durées jusqu'au rachat total en intégrant les données censurées, nous avons donc décidé de tester les 3 modèles du package *xgbse* (*XGBSEDebiasedBCE*, *XGBSEKaplanNeighbours*, *XGBSEKaplanTree*) en les appliquant sur un jeu de données identique afin de pouvoir sélectionner celui proposant les meilleures performances sur notre jeu de test.

A noter que dans ce cas nous prenons en considération la cible de durée complète (définie à la section 2.1.3) comportant les durées de rachat et de censure.

La structure des données est identique à celle utilisée dans nos modélisations précédentes, à savoir que notre base d'entrainement et d'évaluation est constituée sur une répartition 85%/15% des contrats disposant des caractéristiques suivantes :

- Les contrats rachetés en 2017 et souscrits à partir de 2002
- Les contrats rachetés en 2018 et souscrits à partir de 2003

- Les contrats souscrits à partir de 2002 et étant toujours actifs fin 2018 (dans la limite d'une durée de 15 ans pour éviter le biais induit par l'effet d'évolution de la durée des contrats), cette catégorie compose la partie censurée de la base.

Notre base de test elle, est constituée des contrats rachetés en 2019 et souscrits à partir de 2004 ainsi que des contrats souscrits à partir de 2004 et toujours actifs fin 2019 composant la partie censurée de l'échantillon.

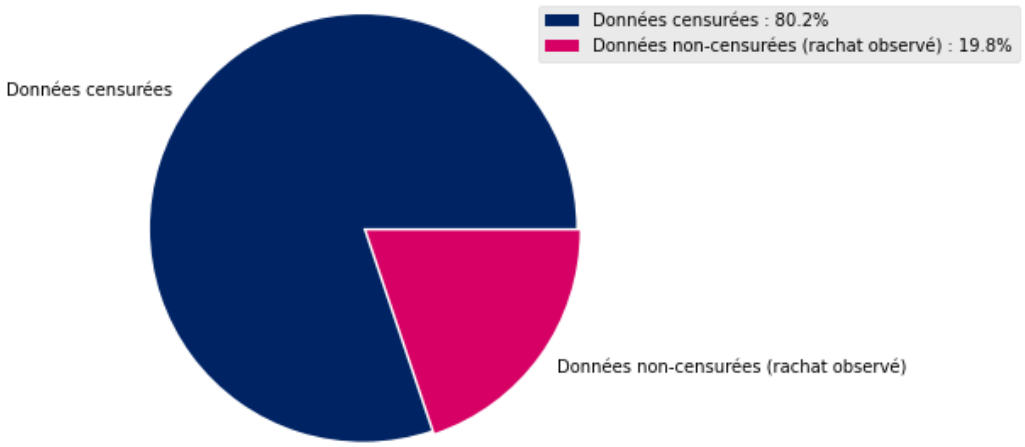

Illustration 27 : Répartition des données censurées dans la base globale

Le graphique précédent montre le ratio entre les données non censurées et les données censurées dans notre base d'origine. On remarque une distribution 80/20 en faveur des données censurées, aisément explicable puisque les événements de rachats sont plutôt rares en comparaison aux nombres de contrats qui restent en activité.

La proportion de données censurées injectées dans les échantillons d'apprentissage, validation et test suivent cette répartition 80/20, c'est-à-dire que pour chaque observation de rachat total, nous avons tiré aléatoirement 4 observations censurées respectant les différentes conditions de

construction des bases d'apprentissage et de test citées auparavant. Et où également aucune des observations ne se retrouve dupliquée dans les différents échantillons.

Il y a aussi un point d'attention à adresser concernant la modélisation. Il faut garder à l'esprit que dans la littérature, les modèles de *Machine Learning* qui permettent l'utilisation de données censurées (comme *xgbse* ou *XGBoost AFT*) utilisent des jeux de données où les observations censurées sont minoritaires, ce qui n'est pas le cas dans notre étude.

Comme pour tout modèle de *Machine Learning*, la calibration des modèles *xgbse* nécessitent d'abord de trouver les valeurs optimales des hyperparamètres. La structure des données n'ayant pas été modifiée, les paramètres trouvés grâce à l'optimisation bayésienne sur les modèles *XGBoost* de régression convenaient à cette application. Pour les paramètres spécifiques aux modèles *xgbse*, tels que le nombre de k-plus proches voisins pour le modèle *XGBSEKaplanNeighbours*, nous avons choisi la valeur de *k* qui maximisait l'indice de concordance sur notre échantillon de validation.

Pour trouver le modèle qui calibre au mieux les courbes de survie sur la base de nos données, nous avons évalué leurs performances sur l'échantillon de test. Les métriques utilisées pour la comparaison sont l'index de concordance et le score de Brier définis précédemment.

	C-index	Score de Brier
Modèle *XGBSEDebiasedBCE*	0.70474	0.08670
Modèle *XGBSEKaplanNeighbours*	0.70105	0.08786
Modèle *XGBSEKaplanTree*	0.67801	0.08803

On remarque que le modèle *XGBSEDebiasedBCE* est le modèle qui détient les meilleures performances sur notre base de test, et donc celui qui est le plus adapté à nos données. A nuancer cependant puisque les différences de performances ne sont pas très significatives, dans le sens où les écarts entre les métriques sont assez faibles.

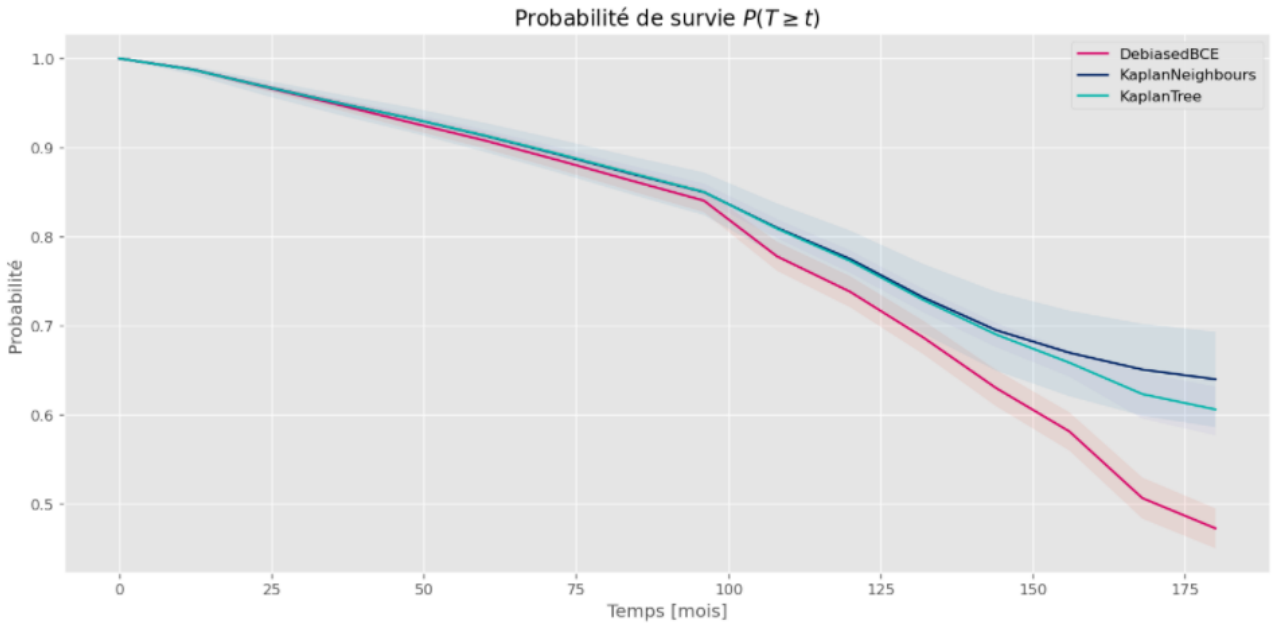

Illustration 28 : Comparaison des courbes de survies moyennes prédites par les modèles

Le graphique ci-dessus compare les courbes de survies agrégées prédites sur l'échantillon de test, ainsi que leur intervalle de confiance. Sur une fenêtre temporelle s'étalant de 0 à 180 mois (15 ans) avec un pas annuel. Les probabilités de survies étant en ce sens calculées par année d'ancienneté.

On observe pour tous les modèles calibrés une chute constante et similaire de la fonction de survie jusqu'au 96ème mois, soit le 8ème anniversaire du contrat, date à laquelle la fiscalité du rachat devient plus souple et où les clients sont plus susceptibles de racheter leurs contrats après cette date. On remarque à cette date une cassure dans la décroissance de la fonction de survie, partagée par tous les modèles et plus particulièrement le modèle *XGBSEDebiasedBCE*, où les probabilités de survie décroissent plus vite.

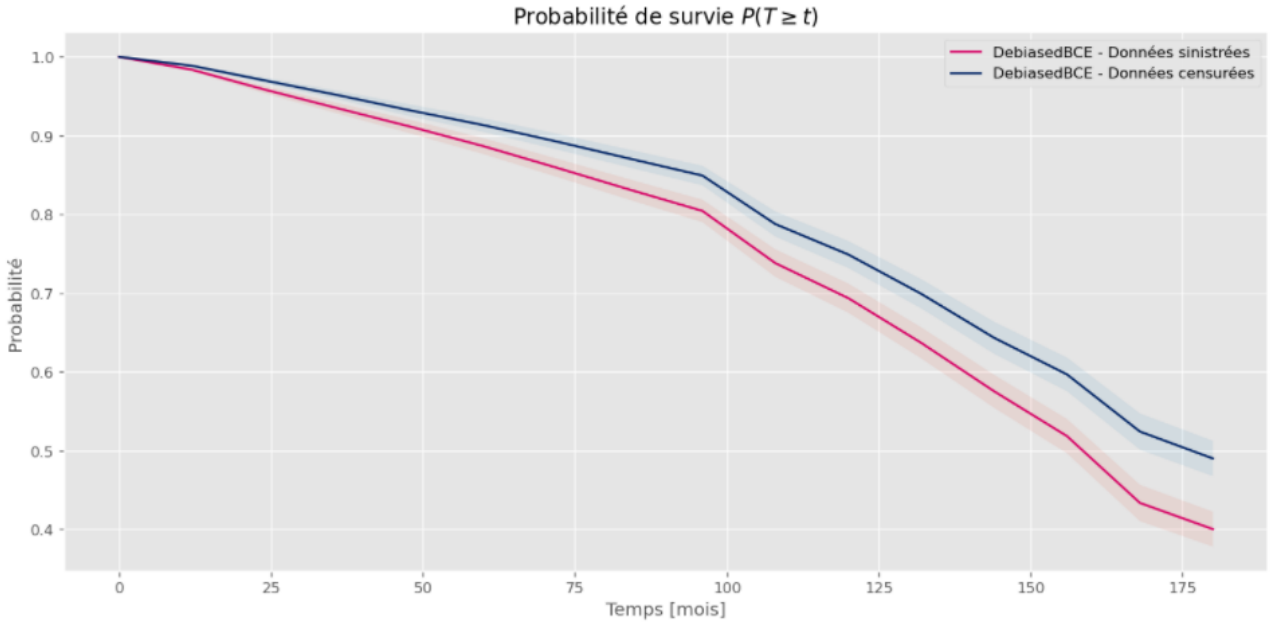

Illustration 29 : Comparaison des prédictions entre les deux populations

Le graphique précédent expose la différence entre les courbes de survies prédites par le modèle *XGBSEDebiasedBCE* sur l'échantillon de test entre deux sous-population : les données sinistrées (en rouge), à savoir les cas où les durées jusqu'au rachat sont connues et les données censurées (en bleu), c'est-à-dire les cas où le rachat n'a pas été observé à sur la période.

On remarque que le modèle arrive à distinguer la plus grande sinistralité des données avec un rachat, avec des probabilités de survie plus faibles à chaque instant t. Cependant, le modèle peine à identifier parfaitement les deux catégories, cela peut s'observer de deux manières différentes.

La première est mesurable grâce à l'index de concordance, un *C-index* autour de 0.7 comme c'est le cas pour tous nos modèles indique bien que l'on arrive à identifier certains profils, mais que le signal reste insuffisant pour pouvoir prédire le moment du rachat de manière efficace pour chaque individu. Le deuxième indicateur est la courbe de survie prédite sur la population censurée. En théorie si le modèle arrivait à discerner parfaitement les profils contrat par contrat, la fonction de survie prédite sur la population censurée dans cette fenêtre temporelle serait une

constante 1 (puisque l'on observe aucun sinistre sur la période). Or ici, la courbe suit la même tendance que celle calculée sur la base des données de sinistres, indiquant que le modèle dispose d'un pourvoir discriminatoire limité.

L'estimateur de Kaplan-Meier avec lequel sont calculées les différentes fonctions de survies repose sur un *embedding* produit par le modèle *XGBoost* en entrée. Etant donné que cet encodeur dispose des mêmes performances que les modèles de régression précédents, où nous avions conclu que les données à disposition (à la souscription du contrat) ne permettent pas à elles seules d'établir une prédiction fine des durées jusqu'au rachat. Les performances des modèles avec censure sont donc cohérentes en rapport aux résultats obtenus précédemment avec les modèles *XGBoost* de régression. La réelle plus-value de l'utilisation des modèles *xgbse* par rapport au modèle *XGBoost* « classique » est de pouvoir calculer des courbes de survies dans le cadre d'étude de durées.

Chapitre

4

Approche n°2 : Modèle de classification

4.1. Cadre de l'approche

Comme vu précédemment, les modèles construits sur la base des données disponibles à la souscription n'apportent pas un signal suffisant. Les modèles comprennent les tendances générales du comportement du portefeuille mais peinent à distinguer les comportements individuels des clients.

4.1.1. Calcul d'un score d'appétence au rachat

Nous changeons ici d'approche, l'idée est de pouvoir prendre en compte les données de la vie du contrat, en ce sens nous utiliserons un modèle de classification d'appétence au rachat total à horizon de temps donné.

Le fonctionnement de la démarche est comme suit : on se place à une date t, parmi les contrats actifs à cette date, on va les séparer en deux catégories :

- Les contrats ayant été rachetés (totalement) au cours du semestre qui suit.
- Les contrats n'ayant pas été rachetés au cours de cette même période.

Le principe de l'apprentissage consiste à donner en entrée du modèle ces deux catégories de contrats avec leurs caractéristiques associées (données à la souscription, données transactionnelles, etc…) dans le but de détecter les spécificités et d'extraire les *patterns* des contrats ayant été rachetés. Par la suite, il s'agit de pouvoir calculer un score sur tout nouveau contrat en rapprochant ses caractéristiques aux *patterns* que le modèle aura précédemment « appris ».

4.1.2. Echantillonnage

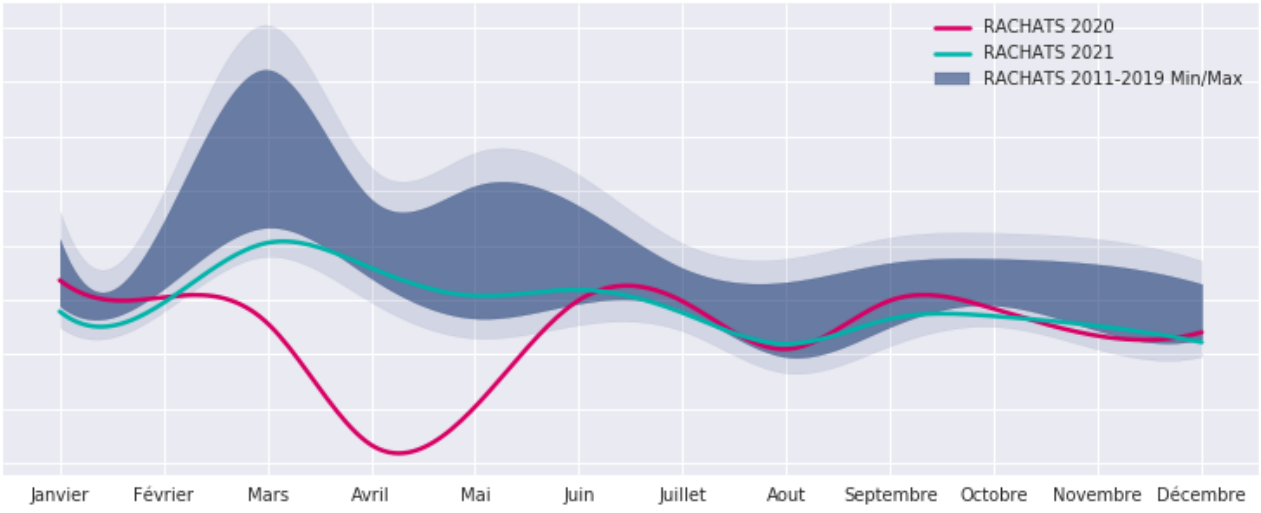

Illustration 30 : Nombre de rachat total par semaine entre 2011 et 2021

Le graphique précédent montre l'évolution du nombre de rachat total de janvier 2011 à Décembre 2021 avec une discrétisation hebdomadaire. La zone en bleu correspond au minimum/maximum observé pour la semaine considérée entre 2011 et 2019, avec une marge de 10% en bleu clair. Les courbes rose et verte correspondent respectivement à l'évolution du nombre de rachat total constaté au cours de l'année 2020 et 2021.

De manière générale, on aperçoit un nombre de rachat plus important au cours des premiers semestres de chaque année, sauf pour l'année 2020.

Sur la période correspondant au premier confinement lié à la pandémie de Covid-19 en France (Mars à mai 2020), on remarque une nette décroissance inédite du nombre de rachat total. De ce fait, nous n'utiliserons pas les deux semestres 2020 dans le cadre de notre étude étant donné l'instabilité des données par rapport aux autres années.

Nous allons donc utiliser l'année 2019 pour l'apprentissage puis passer l'année 2020 et utiliser l'année 2021 pour effectuer le *backtesting* du modèle.

4.2. Construction des nouvelles bases

4.2.1. Architecture de la nouvelle base

Dans cette approche, il s'agit donc de construire 4 jeux de données sur les quatre semestres considérés :

- S1 2019 et S2 2019 pour la base d'entrainement
- S1 2021 pour la base d'évaluation
- S2 2021 pour la base de test

Afin de construire chaque jeu de données, on se place dans un premier temps en début de semestre en récupérant tous les contrats actifs et concernés par le périmètre défini lors de l'approche précédente. Pour rappel il s'agit des contrats appartenant à une sélection de produits d'épargne commercialisés par CNP Assurances.

Pour chacun de ces contrats actifs, sont récupérés les données disponibles à la souscription du contrat ainsi que les données identitaires de l'assuré. Toutes ces données ont été récoltées dans le cadre de notre première approche. Certaines variables comme l'âge de l'assuré en début de semestre, sont à cette occasion mises à jour.

La suite de la section détaillera les données d'inventaire et données d'événements de la vie du contrat qui ont été rajoutées aux données initiales dans le cadre de cette approche, données permettant une analyse plus fine du comportement de l'assuré. Toutes ces données proviennent du même système d'information que l'on a utilisé précédemment.

4.2.2. Définition de la cible (*Target*)

Pour répondre à notre problématique, nous avons besoin d'identifier les contrats ayant été rachetés dans leur totalité au cours du semestre. Ainsi, pour définir notre cible on se place en début de période et on observe les événements de rachat total au cours des 6 prochains mois. On agrège ces événements par contrats pour obtenir le nombre de rachat total sur cette période future.

La cible que l'on va cherche à prédire est donc définie comme suit :

$$CIBLE = \begin{cases} 1 \; si \; on \; observe \; un \; rachat \; total \; sur \; le \; contrat \; au \; cours \; des \; 6 \; prochains \; mois \\ 0 \; sinon \end{cases}$$

4.2.3. Récupération et agrégation des nouvelles variables

Un modèle de *Machine Learning* nécessite de disposer des données sur le passé pour tenter de prédire des événements futurs, il faut donc récupérer pour chaque jeu de données (chaque période) des variables sur l'activité du contrat antérieure à la date de prédiction.

Pour la construction de notre base, on distingue deux types de données :

- Les données d'inventaire (à savoir les données de situation financière à un instant t)
- Les données de transactions (événements sur contrat) agrégées par période

Nous détaillerons ces données dans la suite de cette section.

4.2.3.1. Une double construction des *features*

Les données d'inventaire et les données transactionnelles dépendent d'un contrat, cependant un assuré peut détenir simultanément plusieurs contrats épargne au sein d'une même compagnie d'assurance. Plutôt que de nous restreindre aux données seules du contrat et également pour étendre notre spectre d'information, nous implémenterons chaque variable de deux façons différentes :

- La valeur de la variable sur le périmètre du contrat uniquement.
- La valeur de la variable sur le périmètre de l'assuré, grâce à l'agrégation des données de tous les contrats en sa détention.

On peut schématiser cela de la façon suivante :

Illustration 31 : Exemple de construction d'une variable à vision contrat et à vision client

Soit Monsieur X détenant 3 contrats d'épargne (I, II, III) actifs en début de période. On se place dans le cas où l'on cherche à calculer le score d'appétence au rachat du contrat I. Une des variables utilisées par le modèle concerne l'encours du contrat pris au début du semestre de prédiction.

Comme l'on construit chaque variable d'inventaire et d'évènement sur contrat avec deux visions différentes, on aura donc :

- Variable 1 « Encours du contrat » : L'encours du contrat I (ici 1.000€)
- Variable 2 « Encours du client » : La somme des encours des contrats I, II, III (10.000€)

4.2.3.2. Provisions mathématiques (PM)

Les provisions mathématiques associées aux contrats sont calculées par les actuaires dans le cadre de l'exercice d'inventaire et sont par la suite stockées dans une table de situation financière. Le montant de PM est calculé pour chaque contrat et présente la valorisation monétaire du risque associé jusqu'à la fin du contrat. Ce montant fluctuant avec le temps, les valeurs de PM sont remises à jour semestriellement, ce qui permet de disposer d'un historique.

Nous utilisons cet historique pour pouvoir récupérer à chaque date d'arrêté, les montants de PM pour chaque contrat, ainsi que le détail de répartition de la PM sur les différents supports (en euros ou en unités de compte). Nous récupérons ces valeurs de PM pour la période en cours et la période précédente, pour ainsi pour évaluer l'évolution de la PM pour le contrat et le client au cours du semestre précédent.

Ainsi, les données récupérées permettent de créer les variables suivantes :

- L'encours (PM) du contrat/client au moment de la prédiction
- L'encours « UC » du contrat/client
- L'encours « Euro » du contrat/client
- L'encours du contrat/client au début du semestre précédent
- L'encours « UC » du contrat/client au début du semestre précédent
- L'encours « Euro » du contrat/client au début du semestre précédent
- L'évolution des encours du contrat/client au cours du semestre précédent
- L'évolution des encours du contrat/client sur les différents supports au cours du semestre précédent
- La répartition des encours du contrat/client sur les différents supports

4.2.3.3. Les données de transactions (données événementielles)

Tout flux provenant d'une action du client est enregistré dans une table d'événements actuariels, ainsi on peut extraire dans cette table tout l'historique des transactions effectuées sur un contrat avec comme information le type d'action effectué, sa date, son montant de flux et le support concerné (€/UC). Grâce à cette table on peut récupérer les informations suivantes :

Les versements : Un versement est un flux créditeur venant alimenter le contrat d'assurance de l'assuré. Dans notre étude, cela inclut 3 types de versements :

- Les versements libres, c'est à dire un versement ponctuel par l'assuré, non programmé
- Les versements réguliers, il s'agit de versement programmé avec des montants et des échéances connues à l'avance
- Les versements initiaux, à la souscription du contrat

A partir de ce périmètre, en se plaçant au début du semestre et en observant un historique de 24 mois nous agrégeons ces flux ce qui nous permet de construire les variables suivantes (avec la vision contrat/client) :

- Nombre de versements effectués sur les 12 derniers mois
- Nombre de versements effectués sur les 13 à 24 derniers mois
- Montants des versements effectués sur les 12 derniers mois
- Montants des versements effectués sur les 13 à 24 derniers mois

Les arbitrages : Un arbitrage en assurance vie n'est possible que sur les contrats multisupports, il s'agit d'une action permettant d'effectuer différents types de transfert à l'intérieur d'un contrat :

- D'un support en UC vers un support en euros
- D'un support en euros vers un support en UC
- Ou d'un support en UC vers un autre support en UC

C'est une opération qui peut avoir plusieurs buts : valoriser au mieux son épargne en suivant la tendance du marché, anticiper des fluctuations du marché et donc maximiser le rendement, ou encore modifier sa stratégie d'investissement.

En se plaçant en début de semestre et en observant un historique de 24 mois, on va agréger les événements d'arbitrages (cette fois-ci sur une période unique puisque ces événements sont moins fréquents que les versements) pour construire les variables suivantes (avec la vision contrat/client) :

- Nombre d'arbitrages vers des fonds en euros effectués ces 2 dernières années
- Nombre d'arbitrages vers des supports UC effectués ces 2 dernières années
- Montant des arbitrages vers des fonds en euro effectués ces 2 dernières années
- Montant des arbitrages vers des supports UC effectués ces 2 dernières années

Les rachats : Déjà exposé précédemment, le rachat permet à l'assuré de retirer une partie (rachat partiel) ou la totalité (rachat total) de son épargne.

Dans notre cas, en observant sur un historique de 24 mois les événements de rachats, nous construisons les variables suivantes :

- Nombre de rachats partiels effectués sur le contrat/par le client ces 2 dernières années
- Montants des rachats partiels effectués sur le contrat/par le client ces 2 dernières années
- Nombre de rachats totaux effectués par le client ces 2 dernières années
- Montants des rachats totaux effectués par le client ces 2 dernières années

Pour ces deux dernières variables, l'événement de rachat total étant celui que l'on cherche à prédire et également celui qui clot le contrat, elles ne sont donc construites qu'à une vision client à partir de la constatation de rachat sur un des autres contrats de l'assuré.

4.2.3.4. Données supplémentaires et *feature engineering*

En plus des variables citées précédemment, pour enrichir nos données d'entrées d'autres indicateurs sont construits. Il s'agit de données portant sur le contrat (son ancienneté à date), sur l'assuré : nombre de contrats détenus, ancienneté de ses contrats, distribution des contrats par catégorie de produit (épargne grand public, haut de gamme, retraite…), mais également sur les données de transactions passées puis les variables de montants sont dupliquées et recalculées en proportion de l'encours.

Après avoir obtenu nos 4 jeux de données, nous concaténons les 2 premiers semestres (S1 2019 et S2 2019) afin d'obtenir notre base d'apprentissage, le S1 2021 utilisé comme base de validation et le S2 2021 à l'ensemble de test.

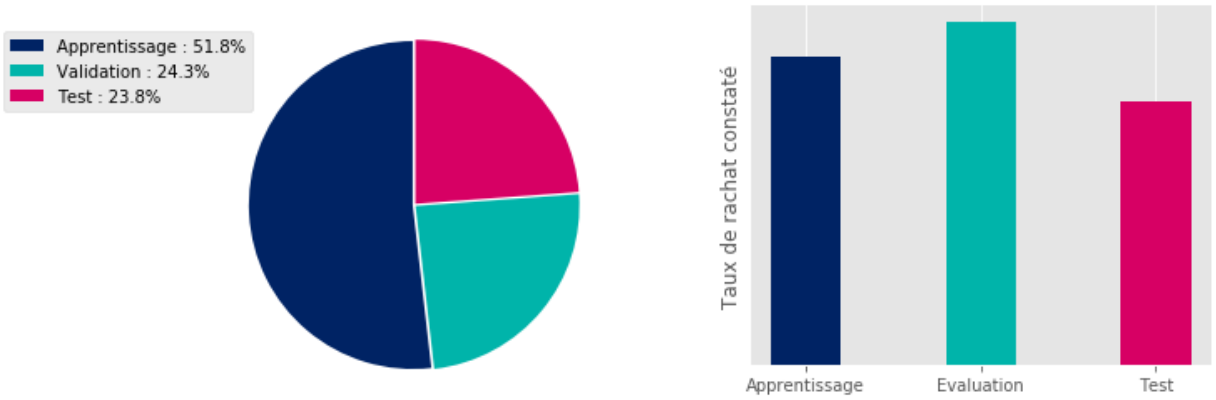

Illustration 32 : Volumes des échantillons et proportion des rachats

Le graphique de gauche représente la proportion en base 100% des observations parmi les jeux d'apprentissage, d'évaluation et de test et celui de droite le taux de rachat observé dans chacun des échantillons.
On remarque une répartition homogène du nombre de données entre les différents semestres, mais une légère différence dans les taux observés. Cela s'explique par les rachats conjoncturels qui sont plus nombreux au cours des premiers semestres de chaque année, également pour 2021 (cf. Ill 30).

4.3. Modélisation et résultats

4.3.1. Modèle utilisé

Pour notre seconde problématique, nous utiliserons un modèle *XGBoost* de classification binaire, choix motivé par la performance et la rapidité de ce modèle, notamment pour la modélisation d'interaction non-linéaires sur un grand ensemble de variables. Au contraire de l'approche précédente par régression, le modèle ici va prédire des valeurs comprises entre 0 et 1 :

- 1 signifiant une appétence forte au rachat total au cours du semestre qui suit
- 0 signifiant une grande aversion au rachat

Pour une classification binaire nous utilisons la fonction de perte logistique implémentée nativement dans *XGBoost*, il s'agit la négation de la fonction de log-vraisemblance moyenne :

$$Log\ loss = -\frac{1}{N} * \sum_{i=1}^{N}[y_i * \log(p_i) + (1 - y_i) * \log(1 - p_i)],$$

avec N la taille de l'échantillon, y_i la valeur de la target observée pour le i-ème contrat (1 si rachat observé, 0 sinon) et p_i la prédiction du modèle pour le i-ème contrat. Cette mesure indique la précision des probabilités prédites par le modèle par rapport à leur valeur observée. Plus les prédictions de probabilités divergent, plus la fonction de perte sera élevée.

Comme expliqué précédemment, les données du S1 et S2 2019 servent à l'entrainement du modèle, après avoir trouvé les méta-paramètres du modèle par optimisation bayésienne et validation croisée. Les données du S1 2021 servent à valider que le modèle n'est pas surentrainé. Les données du S2 2021 quant à elles servent à tester la capacité prédictive du modèle, la période étant déjà passée, les réalisations futures (de rachat ou non) sont déjà connues.

4.3.2. Métriques

4.3.2.1. Courbe ROC et AUC

La courbe ROC (pour *Receiver Operating Characteristic*) permet d'évaluer l'exactitude des prévisions d'un modèle en traçant la sensibilité par rapport au taux de faux positifs (1-spécificité) d'un test de classification.

Graphiquement, on la représente sous la forme d'une courbe qui donne le taux de vrais positifs (fraction des positifs qui sont effectivement détectés) en fonction du taux de faux positifs (fraction des négatifs qui sont incorrectement détectés).

L'aire sous cette courbe (AUC, pour *Area Under Curve*) permet de donner une statistique importante : la probabilité que la prévision d'une observation se trouve dans la catégorie appropriée.

4.3.2.2. Mesure de Lift

Un « Lift » est une mesure de la performance d'un modèle prédictif, mesuré par rapport à un modèle de choix aléatoire. Par exemple, supposons qu'une population ait un taux de rachat prédit égal à 5%, mais qu'un modèle a identifié un sous-groupe avec un taux de rachat prédit de 20%. Ce sous-groupe aura donc un lift égal à 4 (20% / 5%).

Cet indicateur sera produit sous la forme de valeur pour les quantiles de 1% et 10% de l'échantillon possédant les meilleurs scores. Mais également sous forme de courbes, permettant ainsi d'apprécier la performance de prédiction du modèle sur toute la population.

4.3.3. Résultats

Le modèle entrainé donne les résultats suivants :

	Base d'apprentissage	Base de validation	Base de test
AUC	0.786	0.735	0.736
Log-loss	0.122	0.128	0.118
Lift 1%	9.313	5.523	5.893
Lift 10%	3.575	2.680	2.809

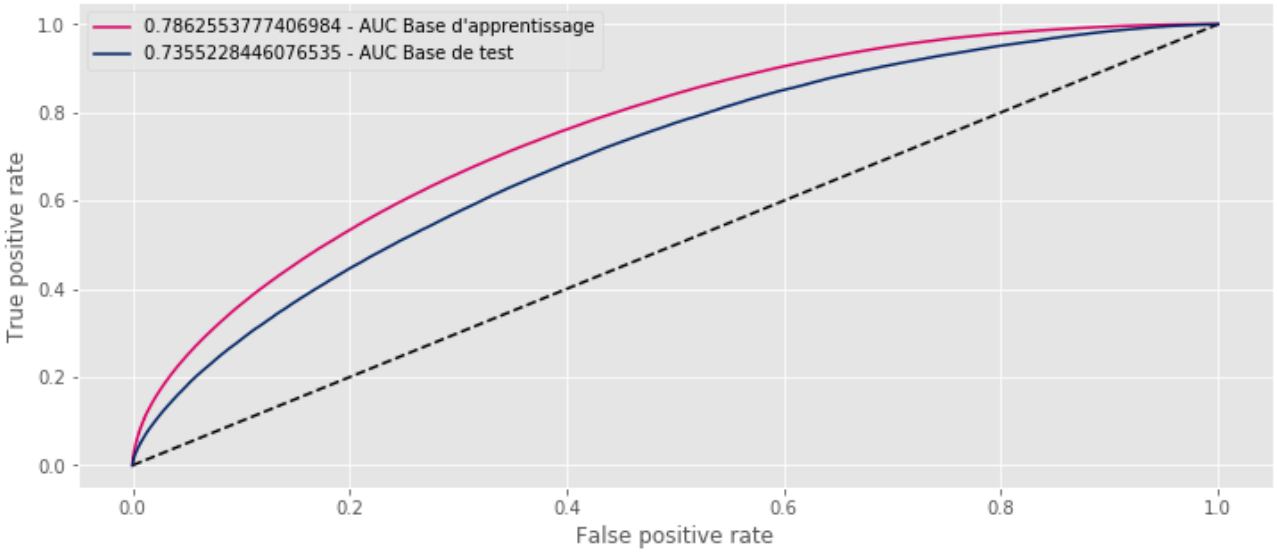

Illustration 33 : Comparaison des courbes ROC sur l'ensemble d'apprentissage et de test

Le tableau et la courbe ROC (avec un AUC de 0.74 sur la base de test) ci-dessus nous indiquent que la qualité de discrimination du modèle est satisfaisante.

Toujours sur la base de test, les valeurs de Lift nous rapportent que le taux de rachat sur nos 1% de meilleurs scores est 5.9 fois supérieur aux taux de rachat observé sur l'ensemble de l'échantillon de test. Cette valeur passe à 2.8 sur les 10% de meilleurs scores.

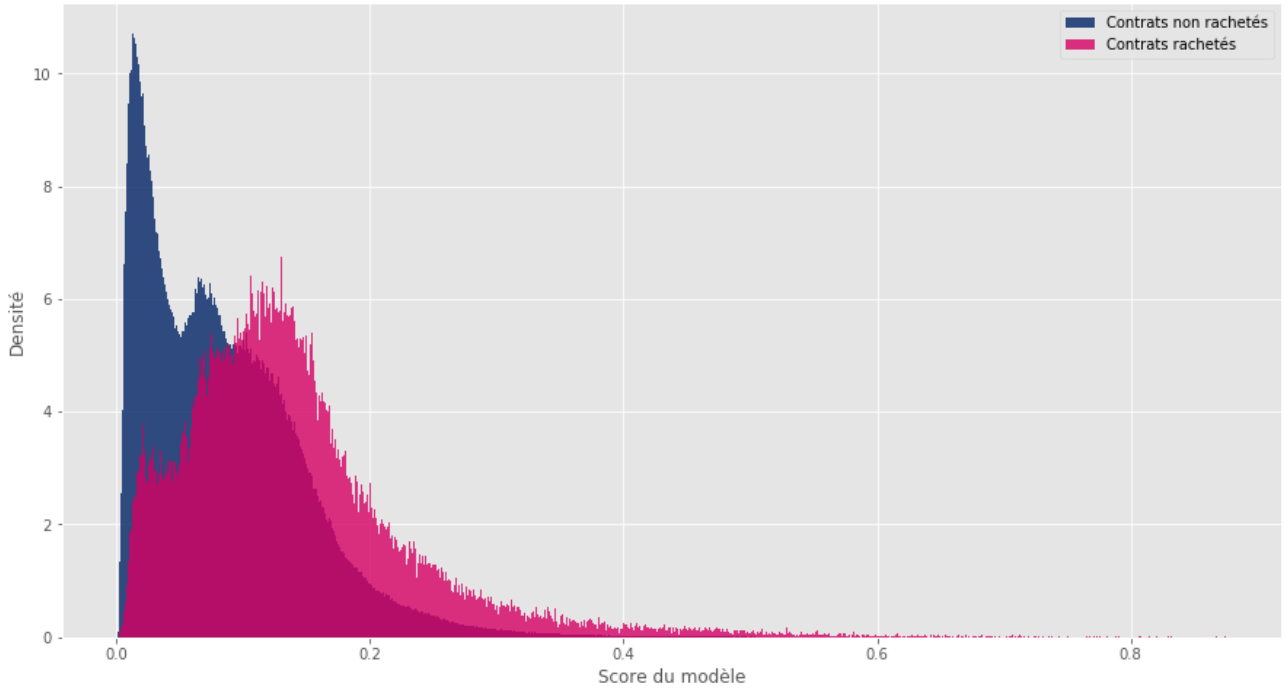

Illustration 34 : Densité des scores sur la base de test

Le graphique ci-dessus représente l'histogramme des scores des prédictions du modèle sur la base de test, avec la distinction entre la sous-population de contrats rachetés (en rose) et les contrats non rachetés (en bleu).

On remarque une densité élevée parmi les scores faibles, avec néanmoins une différence perceptible entre les deux sous-populations. Les scores des contrats rachetés sont de manière générale supérieurs à ceux des contrats non rachetés (un pic de densité à 0.15 pour le premier et une agglomération des scores proches de 0 pour l'autre) ce qui conforte les résultats obtenus précédemment à savoir que le modèle arrive à identifier les deux profils.

Le pic de densité proche de 0 pour les contrats non-rachetés est également une seconde indication de performance du modèle, qui indique que celui-ci arrive bien à discriminer certains profils non appétents au rachat total.

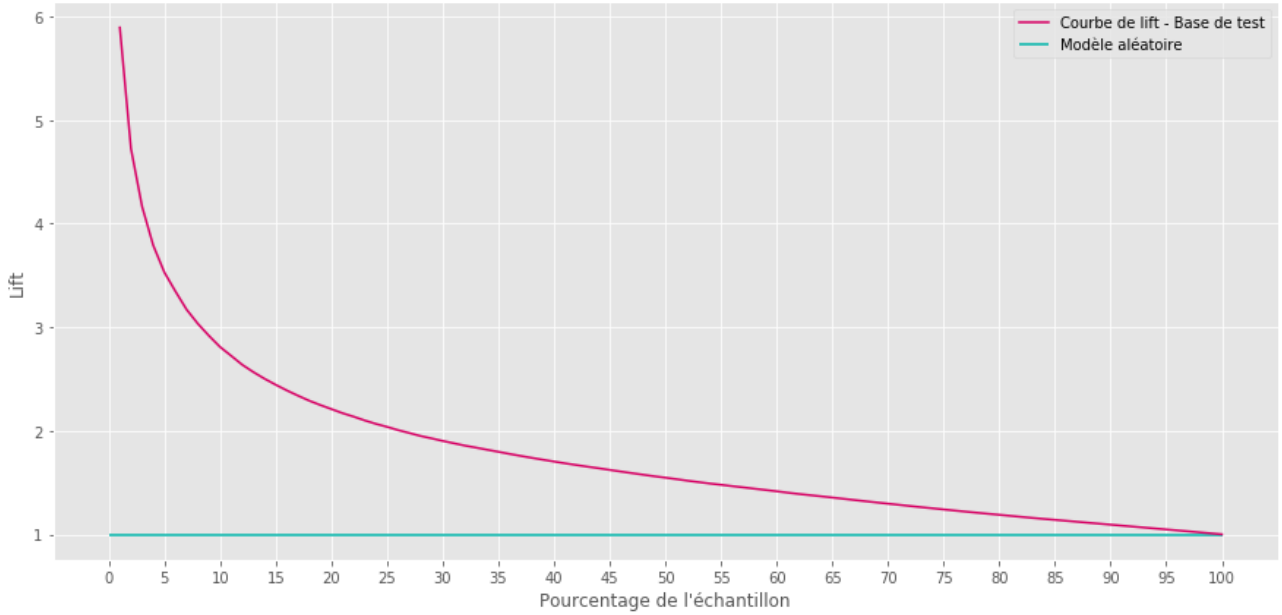

Illustration 35 : Courbe de Lift sur la base de test

Le graphique ci-dessus est la courbe de Lift cumulée, calculée sur la base des prédictions de l'ensemble de test. L'abscisse correspond au pourcentage de l'échantillon considéré, l'ordonnée au facteur de multiplication du taux de rachat observé sur l'ensemble de test.

La ligne verte correspond aux performances d'un modèle aléatoire, par la loi des grands nombres celui-ci dispose d'un taux équivalent à celui de l'ensemble de test, et cela pour n'importe quel sous échantillon. La courbe de Lift en rose correspond au taux de rachat observé parmi les sous-échantillon des x (abscisse) meilleurs scores fournis par le modèle.

Pour les 1% des meilleurs scores, on retrouve bien notre valeur de 5.9 trouvée précédemment, à mesure que l'on inclut des scores plus faibles dans notre sous-ensemble on observe le taux de rachat constaté qui décroit de façon exponentielle, sans saut et sans cassure. Tout cela indique que le modèle arrive à identifier correctement les contrats sur lesquels les clients ont une forte appétence au rachat, et arrive à ordonner correctement les contrats en fonction de leur risque croissant.

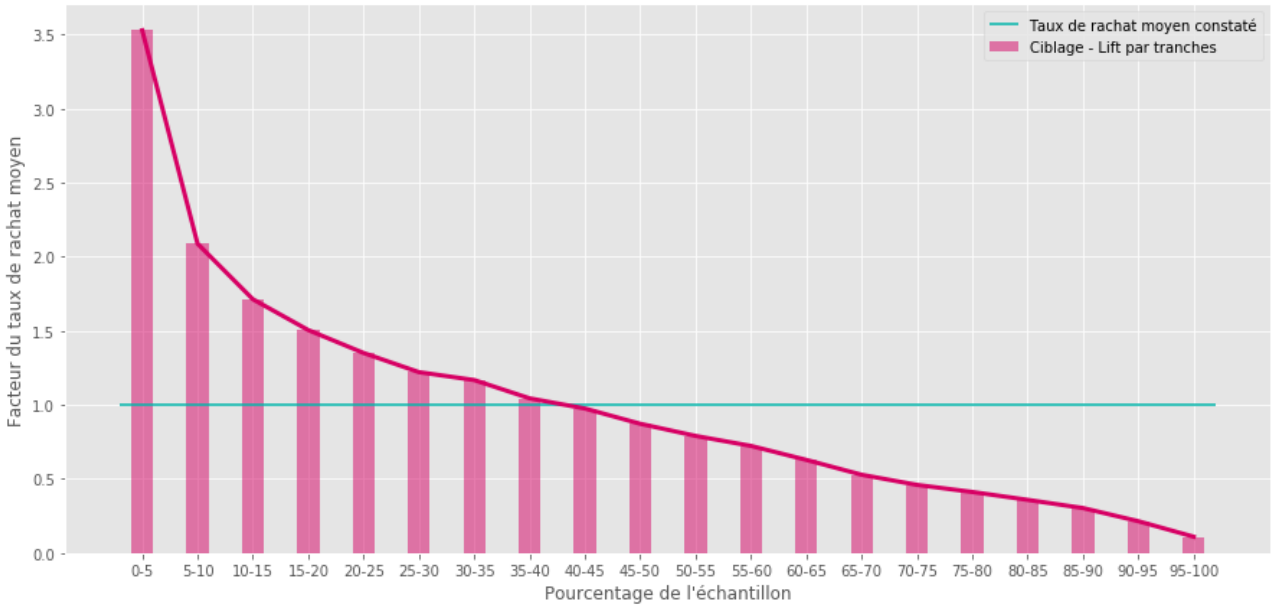

Illustration 36 : Courbe de Lift par tranche sur la base de test

Le graphique précédent est la courbe de Lift calculée par tranche, sur la base des prédictions de l'ensemble de test. L'abscisse correspond aux tranches de scores (allant des 5% des meilleurs scores à gauche jusqu'aux 5% de moins bons scores à droite (95-100%)). L'ordonnée correspond au taux de rachat constaté sur le sous-ensemble composant la tranche, en rapport au taux moyen constaté sur l'échantillon.

La ligne verte (constante 1) correspond au taux moyen sur l'ensemble de test. La courbe et les barres roses correspondent au Lift calculé par tranches de scores.

Cette représentation graphique conforte la conclusion précédente, le modèle arrive à ordonner correctement les contrats en fonction de leur risque d'être rachetés. On voit qu'à mesure que le score de prédiction baisse, le taux de rachat constaté pour les contrats composant la tranche baisse aussi, et cela de façon linéaire.

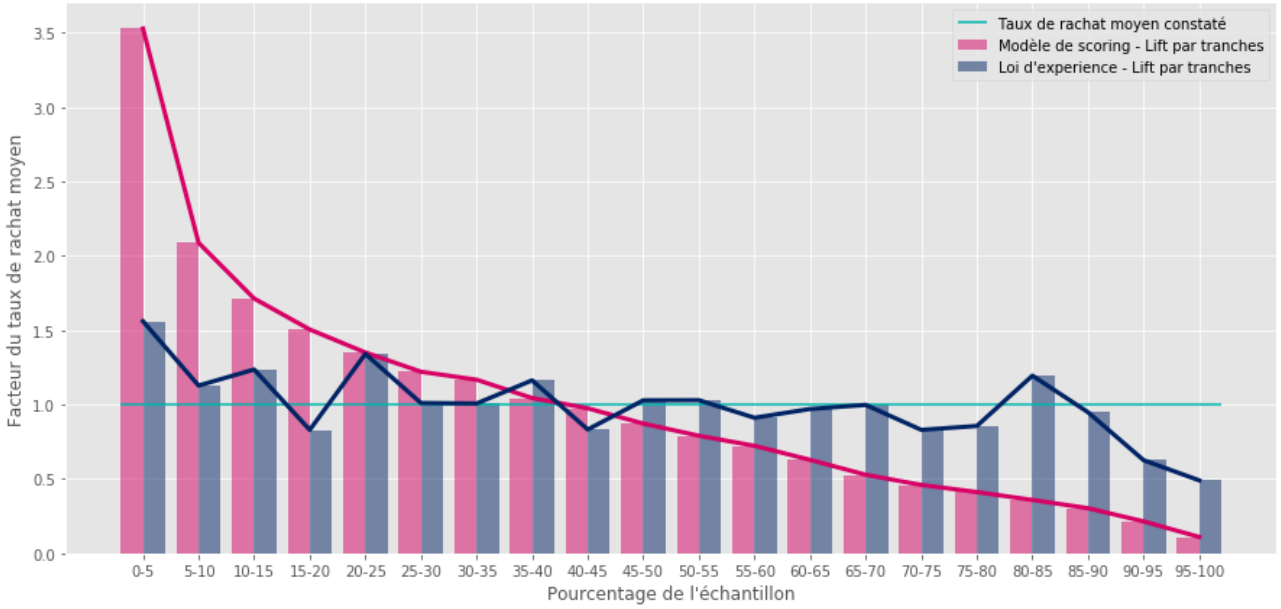

Illustration 37 : Comparaison des Lifts du modèle et des lois d'expérience

Le graphique ci-dessus est une reproduction de la courbe lift précédente construite avec les prédictions du modèle (en rose) et avec comme ajout pour comparaison (en bleu), la courbe lift construite avec les taux de la loi de rachat structurelle du portefeuille épargne. Cette loi de rachat (loi d'expérience) dépend du produit d'épargne considéré et l'ancienneté du contrat.

On remarque que la loi de rachat reflète bien le comportement du portefeuille. Bien qu'elle ne parvienne pas à identifier de manière individuelle les racheteurs, elle donne en revanche le bon taux de rachat au global sur chaque tranche de l'échantillon.

Cependant le modèle de *Machine Learning*, bien que n'ayant pas le même objectif initial par rapport à la construction d'une table d'expérience, est nettement plus performant pour cibler les sous-populations appétentes au rachat à court terme (horizon 6 mois). Sur le premier décile de scores, en comparaison, le modèle ML est au moins 2 fois plus efficace pour détecter les contrats susceptibles d'être rachetés, indiquant ainsi qu'un tel modèle permet d'obtenir des prédictions plus fines, en ayant la capacité de prédire de manière individuelle et donc pour chaque contrat, le risque de rachat associé.

4.3.4. Interprétation

Les algorithmes de M*achine Learning* tels que le *XGBoost* tendent à viser une meilleure performance prédictive au détriment de l'interprétabilité. Pour répondre à cette problématique, une approche appelée SHAP (pour *SHapley Additive exPlanations*) permet d'offrir une interprétabilité globale du modèle. Notamment par la mesure des contributions de chaque *features* dans la construction du score d'appétence au rachat. A l'instar du graphique d'importance des variables, il affiche les variables qui ont le plus grand poids dans la prédiction, mais il informe également des relations positives et négatives de chaque variable avec la cible.

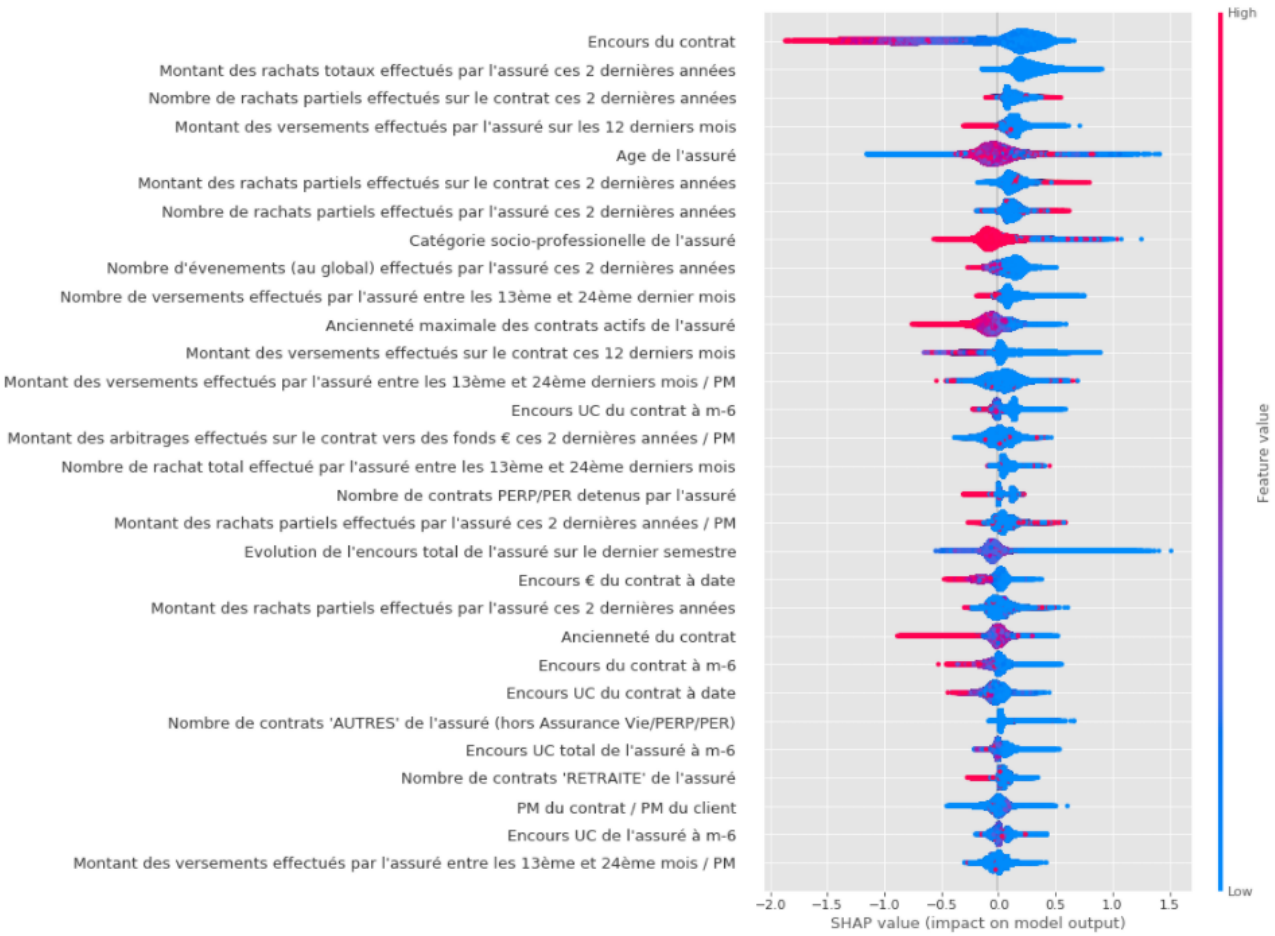

Illustration 38 : Contribution des variables dans la construction du score

Le graphe SHAP précédent construit sur la base de test nous informe des contributions des 30 variables les plus importantes dans la construction du score d'appétence au rachat. Les variables sont listées par décroissance de leur importance, les agglomérations de points rouges correspondent à des valeurs prises par les variables qui sont grandes, les points bleus à des petites valeurs. Des points à gauche correspondent à un impact marginal négatif sur la variable cible, les points à droite à un impact positif sur le score prédit.

On observe certaines variables assez discriminantes, où les zones rouges et bleues sont bien identifiées, permettant ainsi d'avoir une vision claire sur l'impact de la variable, on peut donner quelques exemples :

- L'encours du contrat est la variable qui joue le plus, un client détenant un contrat dont l'encours est faible aura plus tendance à effectuer un rachat total sur celui-ci par rapport à s'il s'agissait d'un contrat avec un encours élevé.

- Un client ayant effectué d'importants versements sur ses contrats au cours des 12 derniers mois aura une tendance à ne pas racheter son contrat à court terme, et inversement.

- L'occurrence de rachats partiels effectués sur le contrat ou effectués par le client sur l'ensemble de ses contrats joue également, plus cette occurrence est importante, plus la probabilité de rachat total à court terme augmente.

- L'impact de l'ancienneté du contrat n'est pas linéaire et son impact dépend en partie des autres variables, cependant on dénote qu'une ancienneté de contrat élevée fait chuter le risque de rachat total du contrat à court terme.

Chapitre

5

Conclusion

Le *Machine Learning* est largement utilisé dans plusieurs secteurs d'activité y compris dans le monde assurantiel qui le voit comme un atout dans le cadre de sa transformation digitale.

Une des interrogations du service s'était portée sur l'apport d'une approche par *Machine Learning* à l'estimation du risque de rachat par rapport à la modélisation classique de ces lois. L'approche avancée dans le cadre de cette étude consistait à estimer de manière individuelle la période où l'assuré serait le plus susceptible de racheter son contrat.

Nous avons fait le choix dans le premier développement d'omettre les informations d'événements de la vie du contrat. Le but était d'évaluer la capacité des modèles de *Machine Learning* à repérer des liens entre les variables explicatives et notre variable cible avec le seul apport des informations disponibles à la souscription.

De la même manière, nous avons fait le choix avec nos premiers modèles de n'utiliser que les cas de rachats totaux avérés, en excluant les cas de censure. Le modèle se concentre uniquement sur la durée à prédire et non sur l'événement ou le non-évènement de rachat. Il s'agit cependant d'un cadre hypothétique et nous avons donc par la suite incorporé les cas censurés. Néanmoins, une telle approche permet de prendre en compte une modélisation rétrospective des durées de rachats. Le modèle est évolutif d'année en année.

Finalement, ces modèles reconnaissent bien les tendances mais peinent à identifier les comportements de rachats. La variance des prédictions s'étend sur 3 ans, ce qui est encore trop vaste pour une utilisation concrète.

Ces travaux ont pu également permettre d'ouvrir de nouveaux sujets en matière de R&D sur la possibilité de personnaliser les fonctions de perte. Certaines problématiques requièrent en effet des solutions qui ne sont pas intégrées nativement dans les algorithmes.

Comme les cas de censures n'étaient pas inclus précédemment et se plaçaient dans un cadre hypothétique, nous avons utilisé des algorithmes récents étant capables de traiter les

problématiques d'analyse de durée de survie, et permettant notamment l'utilisation de données censurées.

A la suite de ce premier développement effectué sur les données disponibles à la souscription, nous avons conclu qu'à elles seules elles n'apportaient pas un signal suffisant pour pouvoir modéliser individuellement le risque de rachat. Nous avons donc élargi le champ de l'étude aux données d'inventaire et de transaction du contrat, afin d'observer le comportement économique de l'assuré.

Dans le deuxième développement, le modèle utilisé a pour but de fournir un score d'appétence au rachat à court terme. Grâce à l'utilisation des données de la vie du contrat comme l'évolution de la provision mathématique annuelle, le nombre de versement, de rachats partiels, les arbitrages effectués, ainsi que leurs montants. Le modèle de *Machine Learning* obtient des résultats très satisfaisants, en ayant la capacité de prédire et d'ordonner correctement les individus en fonction de leur risque de rachat total associé à un instant t.

Cette dernière approche peut être appliquée dans un but de rétention de clientèle ou plus globalement permettre d'optimiser le pilotage du portefeuille. On peut identifier en début de semestre les contrats susceptibles d'être rachetés au cours de celui-ci, puis contacter ces assurés afin de leur faire parvenir une proposition commerciale visant à les conserver au sein du portefeuille.

Ce modèle dispose cependant d'une marge d'amélioration. Parmi les pistes envisagées on peut citer l'implémentation de variables exogènes permettant de suivre les performances des produits d'assurance vie en rapport avec le marché, mais également la prise en compte d'un historique plus important pour les variables événementielles. Tout ceci dans le but de rajouter du signal au modèle et de le rendre plus performant.

Dans un développement ultérieur nous pourrions également appliquer le modèle à tout type d'événements, en particulier celui des rachats partiels, qui ne sont pas inclus dans notre étude.

Chapitre 6

Bibliographie

Webographie

- Fédération Française de l'Assurance, https://www.ffa-assurance.fr/
- France assureurs, https://www.franceassureurs.fr/
- Argus de l'assurance, https://www.argusdelassurance.com/
- Le Revenu, https://www.lerevenu.com/
- https://towardsdatascience.com/
- https://xgboost.readthedocs.io/en/latest/parameter.html
- https://xgboost.readthedocs.io/en/latest/tutorials/aft_survival_analysis.html
- https://xgboost.readthedocs.io/en/latest/tutorials/custom_metric_obj.html
- https://datascientest.com/algorithmes-de-boosting-adaboost-gradient-boosting-xgboost
- https://loft-br.github.io/xgboost-survival-embeddings
- https://github.com/slundberg/shap

Bibliographie

- BARNWALL A., CHO A., HOCKING., (2020). *Survival regression with accelerated failure time model in XGBoost*.

- CHEN T., GUESTRIN C., (2016). *XGBoost, A scalable tree boosting system*.

- CUMMINS J., (1975), *An econometric model of the life insurance sector in the U.S. economy*.

- DOUILLARD M., (2018). *Construction d'un modèle prédictif des comportements d'arbitrage Euro/UC : pertinence de la prise en compte des facteurs psychologiques*. Mémoire d'Actuariat.

- FAUVEL S. et LE PÉVÉDIC M., (2007), *Analyse des rachats d'un portefeuille vie individuelle : Approche théorique et application pratique*, Mémoire d'Actuariat.

- FRIEDMAN J., (2001). *Greedy Function Approximation: A Gradient Boosting Machine.*

- HAIDER H., HOEHN B., DAVIS S., GREINER R., (2020). *Effective Ways to Build and Evaluate Individual Survival Distributions.* Journal of Machine Learning Research 21 (2020) 1-63.

- HAMMING R., (1950). *Error-detecting and error correcting codes.* Bell System Technical Journal 29(2) :147-160.

- HE X. et al., (2014). *Practical lessons from predicting clicks on ads at facebook. Procedings of the 8th International Workshop on Data Mining for Online Advertising (ADKDD'14).*

- HUBERT-CAROL C., (1994). *Durées de vie tronquées et censurées*. Journal de la société statistique de Paris.

- KIM C., (2005). *Modeling surrender and lapse rates with economic variables*, North American Actuarial Journal pp. 56–70. 6.

- KVAMME H., BORGAN Ø., (2019). *The Brier Score under Administrative Censoring : Problem and Solutions.*

- KVAMME H., BORGAN Ø., SCHEEL I., (2018). *Time-to-event prediction with neural networks and Cox regression.* Journal of Machine Learning Research, 20 (129) 1-30.

- LAUR M., (2017). *Anticipation des changements de notes des obligations du portefeuille d'un assureur par méthode de machine learning*. Mémoire d'Actuariat.

- LIN Y., (2006). *Annuity lapse rate modeling: tobit or not tobit?* Society of actuaries 6.

- LUNDBERG S., LEE S., (2017). *A Unified Approach to Interpreting Model Predictions.*

- MARMEROLA G., (2018). *Calibration of probabilities for tree-based models.*

- MARMEROLA G., (2018). *Supervised dimensionality reduction and clustering at scale with RFs with UMAP.*

- MILHAUD X., (2012). *Mélange de GLMs et nombre de composantes :application au risque de rachat en Assurance Vie*, Thèse.

- OUTREVILLE J.F., (1990). *Whole-life insurance lapse rates and the emergency fund hypothesis,* Insurance: Mathematics and Economics 9, 249-255, 6.

- PESANDO J., (1974). *The interest sensibility of the flow of funds through life insurance companies: An econometric analysis*, Journal Of Finance Sept, 1105–1121. 6.

- PLANCHET F., (2020). *Statistiques des modèles paramétriques et semi-paramétriques*. Support de Cours.

- RENSHAW A.E. et HABERMAN S., (1986). *Statistical analysis of life assurance lapses*, Journal of the Institute of Actuaries 113, 459–497. 6.

- SAWYER S., (2003). *The Greenwood and Exponential Greenwood Confidence Intervals in Survival Analysis.*

- TOULET A., (2017). *Modélisation et couverture du risque de rachat total en Epargne Individuelle*. Mémoire d'Actuariat.

- VIEIRA D., MARMEROLA G., GIMENEZ., ESTIMA V., (2021). *xgbse : improving XGBoost for Survival Analysis.*

- YU C. et al., (2011). *Learning Patient-Specific Cancer Survival Distributions as a Sequence of Dependent Regressor.*