Université Toulouse 1 Capitole

MASTER (M2) MENTION ECONOMÉTRIE, STATISTIQUES
PARCOURS STATISTIQUES ET ECONOMÉTRIE

Explorations méthodologiques et opérationnelles autour du processus de construction d'un score

Mémoire professionnel réalisé sous la direction de :

Jean-Philippe KIENNER

Réalisé par :

Saïd OUZZINE

Mémoire présenté et rendu le 23 juin 2024

Table des matières

Liste des figures

Remerciement

Au terme de ce travail, je souhaite exprimer ma profonde gratitude au Professeur Jean-Philippe KIENNER, mon tuteur de mémoire, pour m'avoir accordé la liberté nécessaire à la réalisation de ce projet, tout en y apportant son regard critique et avisé. Je le remercie sincèrement pour son encadrement, ses efforts dévoués, et ses précieux conseils, qui m'ont permis de mettre en pratique les connaissances acquises durant ma formation. Sa méthodologie pédagogique et sa connaissance approfondie du métier ont été inestimables, et je lui adresse ici l'expression de ma plus profonde reconnaissance.

Je tiens également à remercier l'ensemble du personnel de la formation Master 2 Statistiques et Econométrie (FOAD) pour la qualité de l'enseignement dispensé. Un grand merci aussi à tous les membres du corps professoral du Master 2 pour leurs précieux enseignements.

Enfin, je souhaite exprimer ma gratitude aux membres du jury qui examineront et de jugeront ce travail.

Résumé

Ce mémoire explore le développement de modèles prédictifs robustes dans le domaine du Machine Learning, avec un focus particulier sur la stratification des données et le traitement des variables catégorielles. L'objectif principal est de surmonter les défis liés à la complexité croissante et à l'hétérogénéité des données, afin d'améliorer la fiabilité et la performance des modèles prédictifs.

Le projet aborde plusieurs aspects méthodologiques clés :

- ***Stratification des données :*** Assurer une représentation équilibrée dans les ensembles d'entraînement et de test pour une meilleure généralisation des modèles.
- ***Traitement des variables catégorielles :*** Comparer différentes techniques d'encodage pour identifier la plus adaptée.
- ***Modélisation :*** Développer et évaluer plusieurs modèles pour sélectionner les plus performants.
- ***Interprétation des modèles :*** Comprendre les mécanismes sous-jacents pour une prise de décision informée.

En appliquant ces techniques à un projet métier dans le secteur des télécommunications, ce mémoire vise à démontrer l'efficacité des approches de Machine Learning appliqué, particulièrement dans le contexte de la gestion des données complexes et hétérogènes.

Chapitre 1 Introduction générale

Motivation : La motivation pour le choix de ce sujet découle de mon intérêt marqué lors de l'UE Scoring, où j'ai particulièrement apprécié l'exploration approfondie du Machine Learning et de ses objectifs. La méthode de construction de score et les diverses étapes impliquées m'ont captivé, offrant une perspective enrichissante sur l'application des techniques analytiques avancées dans des contextes réels. Ce domaine me fascine pour sa capacité à transformer des données complexes en informations exploitables, permettant ainsi de prendre des décisions éclairées et stratégiques. L'enseignement de cet UE Scoring m'a incité à consolidés mes connaissances et compétences dans le domaine du Machine Learning appliqué à la gestion et à l'analyse de données, et à explorer comment ces techniques peuvent être optimisées pour répondre efficacement aux défis spécifiques rencontrés par les entreprises, comme celui de la fidélisation des clients dans le secteur des télécommunications.

1 Problématique et objectif du projet

Le développement de modèles prédictifs robustes rencontre des défis importants liés à la complexité croissante des données et à leur hétérogénéité. Une stratification précise des données est essentielle pour assurer une représentation équilibrée dans les ensembles d'entraînement et de test, garantissant ainsi une généralisation adéquate du modèle. De plus, le traitement efficace des variables catégorielles est crucial pour éviter les problèmes de sur-dimensionnalité ou de perte d'information, impactant ainsi la performance des modèles prédictifs. La sélection et la comparaison des approches de modélisation sont également essentielles pour évaluer la performance et la fiabilité des prédictions. Enfin, une interprétation approfondie des modèles est indispensable pour clarifier les décisions prises par des modèles complexes et définir un score prédictif final, nécessaire à une prise de décision informée.

2 Déroulement du projet

Ce mémoire se déroulera en plusieurs étapes méthodologiques clés visant à garantir la robustesse des résultats obtenus. La stratification minutieuse des données assurera une répartition équilibrée des catégories dans les ensembles d'entraînement et de test, favorisant ainsi une généralisation appropriée du modèle prédictif. Le traitement des variables catégorielles fera l'objet d'une attention particulière, explorant différentes techniques d'encodage pour identifier la plus adaptée. La phase de modélisation impliquera le développement et la comparaison de plusieurs modèles prédictifs afin de déterminer la méthode la plus efficace. Enfin, une analyse approfondie des modèles permettra de comprendre les

mécanismes sous-jacents aux prédictions, enrichissant ainsi la littérature académique dans ce domaine.

Pour ce projet de mémoire, les données fournies durant l'UE scoring du Professeur KIENNER seront utilisées. Ces données comprennent les caractéristiques socio-démographiques des clients, les détails sur les produits utilisés et l'historique des résiliations jusqu'au 31 mars 2023. Elles serviront à appliquer différentes approches dans le développement du modèle prédictif, en extrayant une cohorte de 2000 clients pour l'analyse. L'historique des résiliations sera essentiel pour l'entraînement et l'évaluation des performances des modèles, facilitant ainsi la conception d'une stratégie de fidélisation précise et efficace grâce à une analyse approfondie quantitative et qualitative.

3 Description générale des données

Dans cette section, je présenterai un résumé décrivant les données Telecom utilisées ainsi que les étapes de prétraitement appliquées durant de l'UE Scoring. L'objectif est d'assurer au lecteur une compréhension claire du contenu des données utilisées dans ce mémoire, ainsi que de présenter l'objectif métier lié à ces données.

3.1 Prétraitement des données

La base de données utilisée regroupe les clients d'un opérateur de téléphonie mobile jusqu'au 31 mars 2023, fournissant des détails sur leurs caractéristiques socio-démographiques, leurs produits et leur utilisation. Pour identifier une cohorte de 2000 clients, cette base sera exploitée. Une seconde base inclut une colonne supplémentaire indiquant si le client a résilié entre le 1er janvier 2023 et le 31 mars 2023. Ces données seront utilisées pour développer des modèles statistiques.

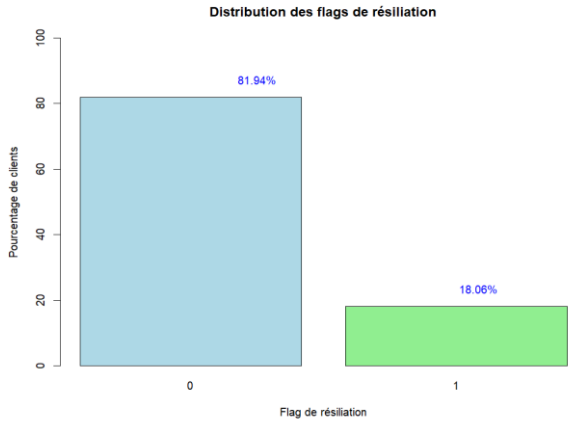


Figure 1: Distribution des flags de résiliation

- La classe 0, qui correspond aux clients n'ayant pas résilié leur contrat, représente environ 81.94% de l'ensemble des clients.

- La classe 1, qui correspond aux clients ayant résilié leur contrat, représente environ 18.06% de l'ensemble des clients.

Cela suggère que la majorité des clients ne résilient pas leur contrat, tandis qu'une minorité résilie effectivement. Il est important de noter cette disparité lors de l'analyse des modèles prédictifs et des stratégies de rétention.

Pour initier le prétraitement des données, les variables de date ont été converties au format Date dans la base de données. Cela inclut les colonnes relatives à la date de naissance, la date d'activation, la date de fin d'engagement, et la date du dernier réengagement. Cette transformation vise à garantir une manipulation précise des dates lors des analyses ultérieures, assurant ainsi l'exactitude et l'efficacité des opérations telles que les calculs de durées ou les comparaisons.

3.2 Nettoyage des Données

Ensuite, un nettoyage des données a été réalisé pour gérer les valeurs manquantes. Les colonnes affectées ont été identifiées en calculant le nombre de valeurs manquantes par colonne, puis affichées. Différentes méthodes d'imputation ont été appliquées pour remplacer les valeurs manquantes par des estimations, améliorant ainsi la qualité et la complétude des données et rendant les analyses ultérieures plus fiables.

- **Imputation des Dates de Naissance**

Pour imputer les dates de naissance manquantes, la médiane a été utilisée en stratifiant par groupe de catégorie socio-professionnelle (CSP). Cela signifie que pour chaque CSP, les dates de naissance manquantes ont été remplacées par la médiane des dates de naissance du groupe correspondant. Cette approche utilise une valeur centrale moins sensible aux valeurs extrêmes, et la stratification par CSP prend en compte les variations démographiques spécifiques à chaque groupe, assurant ainsi une imputation précise et contextuellement adaptée.

- **Imputation du Nombre de SMS**

Le nombre de SMS envoyés trois mois avant la date de référence (nb_sms_m3) a été imputé en utilisant la moyenne des SMS envoyés sur les mois non manquants (mois 1, 2, 4, 5 et 6) pour chaque client. Cette approche personnalisée assure une imputation précise en utilisant les données disponibles spécifiquement pour chaque client. Après cette imputation, la colonne temporaire utilisée pour le calcul de la moyenne a été supprimée, finalisant ainsi le processus de nettoyage des données. Cette méthode d'imputation optimise l'utilisation des informations disponibles tout en préservant l'intégrité des analyses statistiques et en réduisant les biais liés aux données manquantes.

3.3 Enrichissement des Données

Pour enrichir les données et améliorer les analyses prédictives, plusieurs nouvelles variables ont été créées en utilisant la date de référence du 31 décembre 2022. Ces variables offrent des informations fondamentalles sur les clients et leur comportement, contribuant ainsi à une meilleure identification des clients à risque de résiliation.

- **Calcul de l'Âge des Clients**

L'âge des clients a été calculé à partir de leur date de naissance, offrant une vue démographique essentielle pour le modèle.

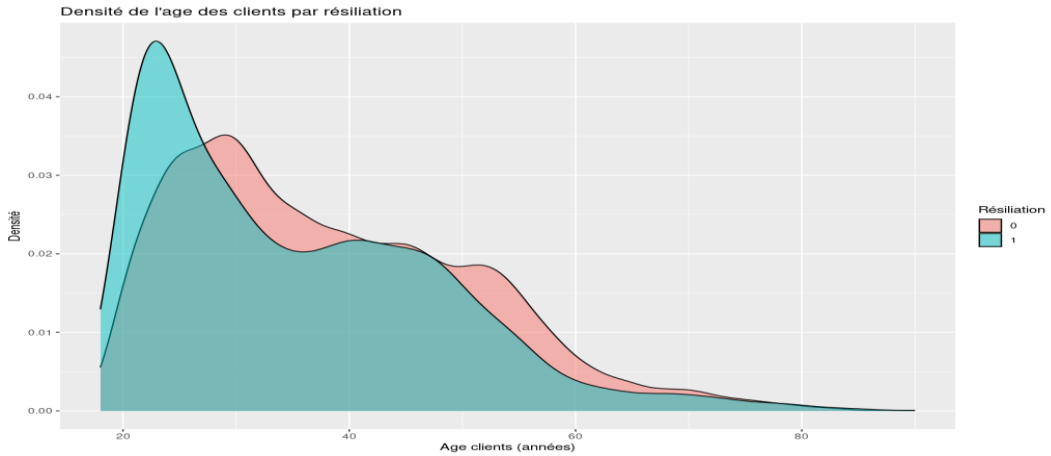


Figure 2 : Densité de l'âge des clients par résiliation

Le graphique de densité montre que les jeunes adultes, vers 20 ans, représentent une proportion plus élevée des clients, avec une densité plus marquée. Les clients ayant résilié leurs contrats sont plus fréquents parmi les jeunes adultes (20-30 ans) comparé aux non-résiliés. Au-delà de 30 ans, la densité des deux groupes se rapproche, bien que les non-résiliés restent légèrement plus nombreux jusqu'à environ 50 ans. La densité diminue avec l'âge pour les deux groupes, particulièrement chez les clients résiliés, suggérant une résiliation moins fréquente chez les clients plus âgés. En résumé, les jeunes adultes ont tendance à résilier plus souvent, tandis que les clients plus âgés montrent une plus grande fidélité aux contrats.

- **Calcul de l'Ancienneté des Clients**

L'ancienneté des clients, mesurée depuis leur date d'activation, fournit une perspective sur leur durée de relation avec l'opérateur, ce qui peut être un indicateur clé de fidélité ou de risque de churn.

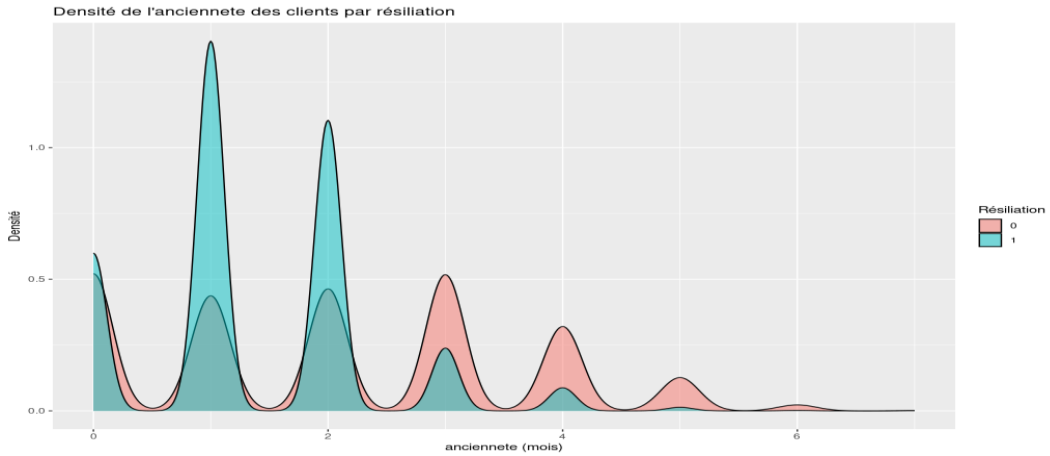


Figure 3 : Densité de l'ancienneté des clients par résiliation

Le graphique de densité montre des pics distincts d'ancienneté des clients en fonction de la résiliation. Les clients résiliant leur contrat présentent des pics élevés à environ 0, 1 et 2 mois, suggérant des résiliations fréquentes peu après l'adhésion, surtout au premier mois. En revanche, les clients non résiliant montrent une densité plus constante mais plus faible, avec un léger pic vers 3-4 mois, indiquant une moindre probabilité de résiliation avec une plus grande ancienneté. En résumé, les résiliations sont fréquentes peu après l'adhésion, tandis que les clients restants plus longtemps sont plus susceptibles de maintenir leur contrat.

- **Durée d'Engagement Restante**

La durée d'engagement restante a été déterminée, indiquant combien de temps il reste aux clients avant la fin de leur contrat actuel. Les clients dont l'engagement touche à sa fin peuvent être plus susceptibles de résilier, faisant de cette variable un facteur déterminant dans le modèle prédictif.

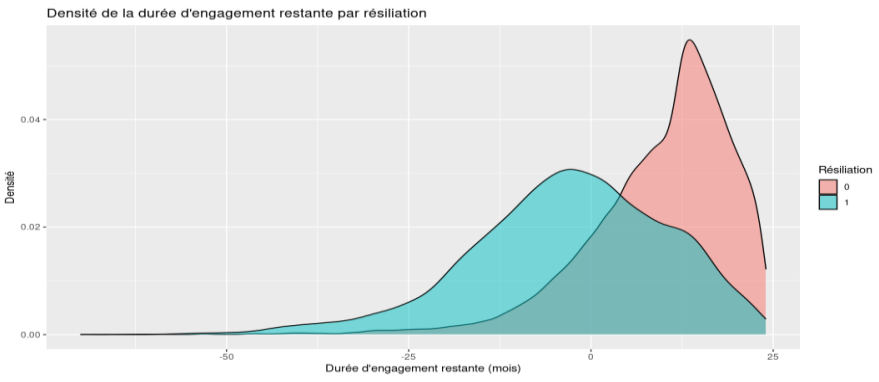


Figure 4 : Densité de la durée d'engagement restante par résiliation

- **Ancienneté Depuis le Dernier Réengagement**

L'ancienneté depuis le dernier réengagement a également été calculée. Cette variable montre combien de temps s'est écoulé depuis que les clients ont renouvelé leur contrat pour la dernière fois, offrant une perspective sur leur engagement récent avec l'opérateur. Les valeurs

manquantes de cette variable ont été imputées par la moyenne de la strate d'engagement correspondante, garantissant ainsi la cohérence des données.

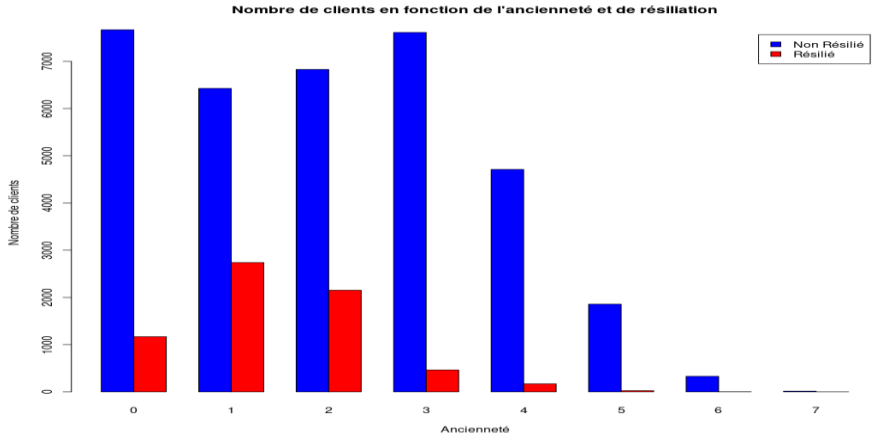


Figure 5 : Nombre de clients en fonction de l'ancienneté et de résiliation

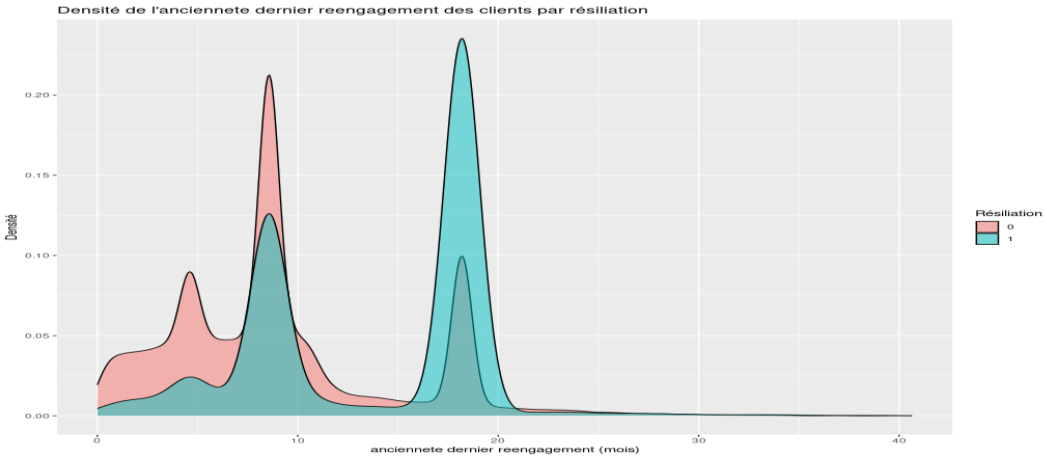


Figure 6: Densité de l'ancienneté dernier réengagement des clients par résiliation

Le graphique de densité montre deux pics distincts pour l'ancienneté du dernier réengagement des clients en fonction de la résiliation. Les clients ayant résilié montrent des pics à environ 8 et 20 mois, suggérant deux périodes critiques où la résiliation est plus probable. En revanche, les clients qui n'ont pas résilié présentent un pic autour de 10 mois, indiquant une période intermédiaire moins propice à la résiliation. Ces observations suggèrent que la résiliation tend à se produire peu de temps après le réengagement ou après une période prolongée, ce qui peut orienter les stratégies de rétention en ciblant ces moments critiques.

Chapitre 2 Stratification des données

Dans le cadre de notre analyse, la stratification des données constitue une étape cruciale pour garantir la qualité et la pertinence des résultats obtenus. Ce chapitre se propose des stratégies de stratification permettant de segmenter les données de manière significative.

1 Principe de Stratification des Données

La stratification des données est une technique essentielle dans les projets de modélisation prédictive, surtout lorsque les classes cibles sont déséquilibrées comme dans notre cas et comme c'est souvent le cas dans les problèmes de churn ou de résiliation. Dans ce projet, la stratification des données vise à garantir une distribution équilibrée des classes dans les ensembles d'entraînement et de test, ce qui est important pour une évaluation fiable et une bonne généralisation du modèle *(Breiman, L,2001)*.

La stratification des données offre de nombreux avantages en termes de représentativité, de réduction de la variance et d'amélioration des performances du modèle *(Hastie, T,2009)*, elle peut également présenter des défis en termes de complexité et de sélection des strates appropriées. Il est donc crucial de bien comprendre les caractéristiques des données et les objectifs du modèle avant de décider d'appliquer une stratification *(Kuhn, M., & Johnson, K. ,2013)*.

- **Avantages de la Stratification**
  - ***Assurer la Représentativité :*** La stratification garantit que toutes les classes d'intérêt sont représentées de manière proportionnelle dans les échantillons, évitant ainsi les biais potentiels dans le modèle.
  - ***Réduire la Variance :*** En réduisant la variance des estimations statistiques, la stratification améliore la précision et la stabilité des modèles prédictifs.
  - ***Améliorer les Performances du Modèle :*** Une distribution équilibrée des classes dans les échantillons permet au modèle d'apprendre de manière plus efficace les relations sous-jacentes, conduisant à des prévisions plus précises et à des performances globalement meilleures.
- **Inconvénients de la Stratification**
  - ***Complexité de définition :*** La stratification peut nécessiter des efforts supplémentaires pour identifier et définir les strates pertinentes, en particulier dans les ensembles de données complexes avec de nombreuses variables catégorielles.
  - ***Perte de généralisation :*** Une stratification inadéquate peut conduire à une perte

de généralisation si les strates ne sont pas correctement définies ou si elles ne capturent pas les variations importantes dans les données.

- ***Complexité de modélisation :*** La stratification peut entraîner une augmentation de la complexité de la modélisation et de l'interprétation, en particulier lorsque les données sont fortement déséquilibrées ou lorsque les strates sont mal définies.

2 Choix de la variable de stratification

Dans le cadre de la prédiction résiliation, la sélection de la durée d'engagement restante comme critère de stratification repose sur plusieurs considérations cruciales, premièrement, cette mesure est intrinsèquement liée à l'objectif du projet, car elle reflète la période durant laquelle un client reste sous contrat. Les clients avec une durée d'engagement restante plus courte présentent souvent un risque accru de résiliation, étant moins liés contractuellement et plus enclins à rechercher des alternatives. En stratifiant les données selon cette durée, le modèle peut mieux appréhender les variations de comportement et identifier les segments de clients à risque de churn. De plus, la durée d'engagement restante est une variable facilement calculable à partir des données contractuelles existantes, facilitant ainsi son intégration pratique dans le processus de stratification. Ainsi, en optant pour cette variable comme critère de stratification, le modèle peut prendre en compte avec précision le niveau de risque de churn associé à chaque segment de clients, améliorant ainsi la qualité et la fiabilité des prédictions.

3 Choix de seuil et de clustering

Une fois que la durée d'engagement restante a été calculée et employée comme base de stratification des données, la question primordiale est de déterminer les seuils de stratification. Notre approche initiale pour répondre à cette question a été de visualiser graphiquement la densité des histogrammes afin d'observer la distribution des résiliations en fonction de cette variable.

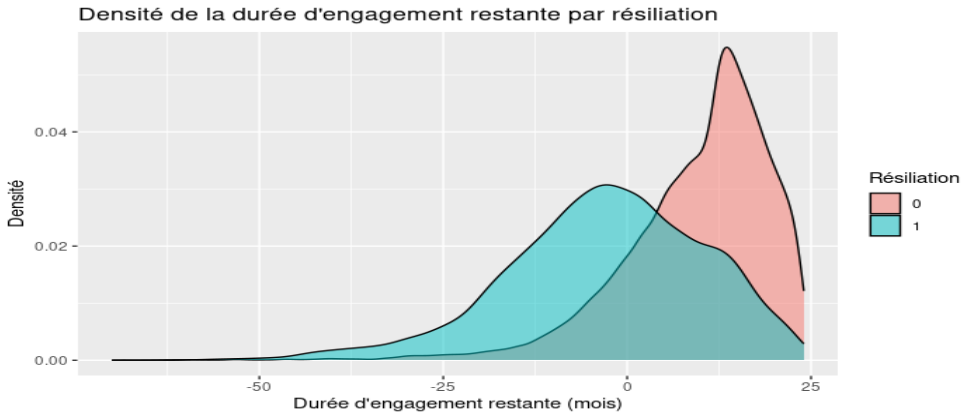


Figure 7 : Densité de la durée d'engagement restante par résiliation

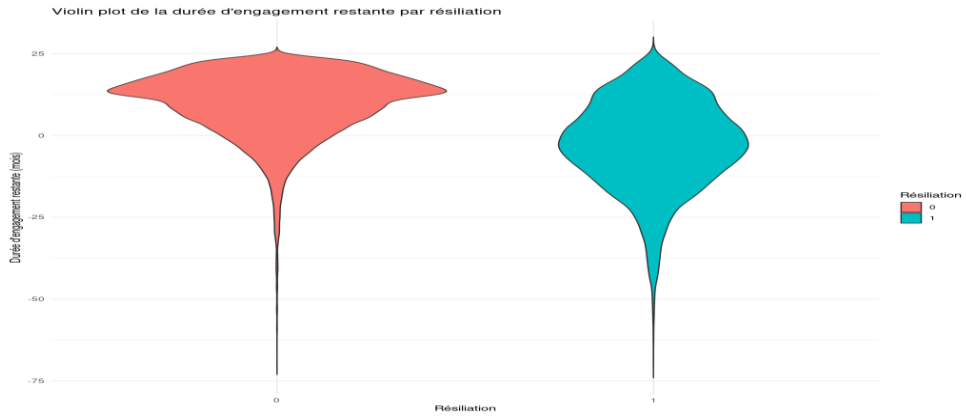


Figure 8 : Violin plot de la durée d'engagement restante par résiliation

Le graphique de densité illustre la répartition de la durée d'engagement restante (en mois) pour les clients résiliés et non résiliés, offrant un aperçu pour l'identification des clients à risque de résiliation. Malgré un chevauchement significatif des distributions, une tendance émerge : les clients résiliés tendent à être concentrés vers des durées d'engagement restantes plus courtes, suggérant un lien potentiel entre une durée d'engagement réduite et le risque de résiliation.

L'absence d'un seuil distinct sur l'axe des abscisses (x) complique la séparation entre les deux groupes dans le graphique de densité. Cette complexité souligne l'utilité de recourir à l'approche du clustering (Jain, A. K, 2010), notamment à travers l'algorithme ***K-means***. Ce dernier vise à segmenter les clients en clusters homogènes en fonction de leurs caractéristiques, ce qui permet d'identifier des sous-groupes de clients présentant des similitudes. Dans notre cas, le ***K-means*** pourrait être employé pour regrouper les clients en fonction de leur durée d'engagement restante, offrant ainsi des informations supplémentaires sur les segments de clients à risque de résiliation.

Pour déterminer le nombre optimal de clusters, nous avons utilisé *la méthode du coude*. Cette méthode itère à travers un éventail de valeurs pour le nombre de clusters (de 2 à 10, dans notre cas pour une première exploration) (Kaufman, L., 2009), applique l'algorithme K-means à

chaque nombre de clusters, puis calcule la somme des carrés intra-cluster (Within Sum of Squares) à chaque itération. Ensuite, un graphique de la somme des carrés intra-cluster en fonction du nombre de clusters est tracé pour visualiser le point où l'ajout de clusters supplémentaires ne réduit plus de manière significative la somme des carrés intra-cluster, aidant ainsi à déterminer le nombre optimal de clusters (Tibshirani, R2001).

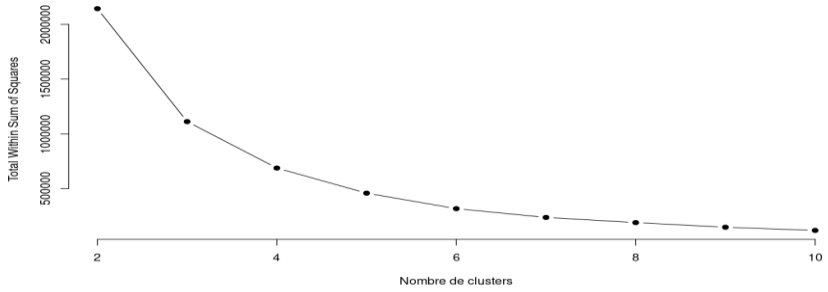


Figure 9 : Graphique pour déterminer le nombre optimal de clusters

Le graphique présenté montre le total des carrés des sommes intra-groupes (Within Sum of Squares, WSS) en fonction du nombre de clusters (de 2 à 10) pour l'algorithme de K-means appliqué à la variable "duree_engagement_restante_mois".

L'objectif de la méthode du coude est d'identifier **le nombre optimal de clusters** en se basant sur la courbe du WSS. Le principe est de choisir le nombre de clusters où la diminution du WSS devient moins importante, ce qui suggère que l'ajout de clusters supplémentaires n'apporte plus d'amélioration significative à la qualité du regroupement. Dans notre cas, on observe une **diminution importante du WSS** entre 2 et 3 clusters, suivie d'une diminution plus progressive par la suite. Cela suggère que le **nombre optimal de clusters pourrait être de 3**.

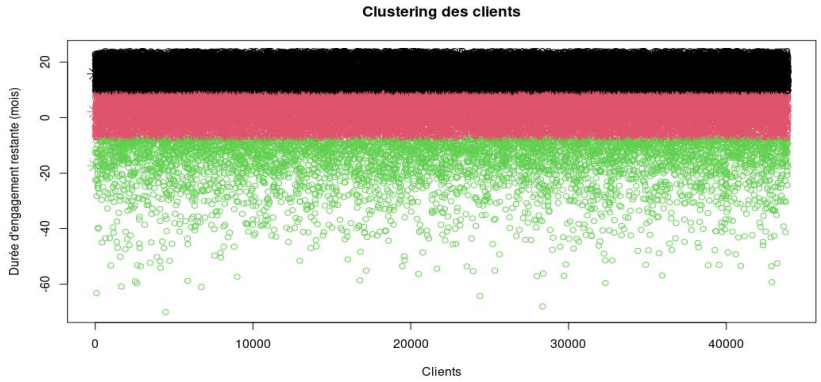


Figure 10 : Clustering des clients

4 Description des différentes states

En utilisant l'algorithme K-means avec trois clusters distincts, les segments suivants ont été identifiés :

Cluster 1 : Ce segment peut être caractérisé comme les clients "**fidèles**", avec une durée d'engagement restante moyenne de plus de **15 mois**. Il présente le **taux de résiliation le plus bas, avec seulement 7% de clients résiliés**.

Cluster 2 : Ce groupe représente les clients "**intermédiaires**", avec une durée d'engagement restante moyenne **d'environ 2 mois**. Bien que leur risque de résiliation soit plus élevé que celui du cluster 1, il reste modéré, avec environ **23% de clients résiliés**.

Cluster 3 : Ce cluster regroupe les clients ayant déjà résilié leur contrat. Il se distingue par un taux de résiliation significativement plus élevé, avec environ **56% de clients résiliés**.

Cluster	Caractéristiques	Durée d'engagement restante moyenne	Taux de résiliation
Cluster 1	Clients fidèles	Plus de 15 mois	7% de clients résiliés
Cluster 2	Clients intermédiaires	Environ 2 mois	Environ 23% de clients résiliés
Cluster 3	Clients ayant résilié	N/A (clients ayant déjà résilié)	Environ 56% de clients résiliés

- **Évaluation de ce clustering**

Pour évaluer la validité des clusters générés par l'algorithme *K-means*, nous utilisons plusieurs mesures de performance des clusters. Voici une explication des indices utilisés et les seuils de référence tirés de la littérature pour déterminer la qualité du clustering :

- **Coefficient de silhouette :** Mesure à quel point chaque point de données est similaire à son propre cluster par rapport aux autres clusters.
  - ***Valeur comprise entre -1 et 1 :*** une valeur proche de 1 indique des clusters bien séparés et compacts, une valeur proche de 0 indique que les points se trouvent sur ou très près des limites des clusters, et une valeur négative indique que les points ont été mal attribués à des clusters incorrects.
  - ***Seuil de référence :*** Un coefficient de silhouette supérieur à 0,5 est généralement considéré comme une indication de clusters bien formés.

  Résultat : Une moyenne de coefficient de silhouette de 0,577, suggérant une bonne séparation et une bonne cohésion des clusters.
- **BIC (Bayesian Information Criterion) pour le clustering via un modèle de mélange gaussien :** Critère d'information bayésien en français, est une mesure statistique utilisée pour évaluer différents modèles probabilistes, notamment dans le contexte de l'analyse de clusters. Il est souvent utilisé pour comparer la qualité de modèles de clustering lorsque l'on cherche à déterminer le nombre optimal de clusters.

$$BIC = log(n).p - 2.log(\hat{L})$$

Où : *n est la taille de l'échantillon* ; *P est le nombre de paramètres du modèle et $\hat{L}$ est la valeur maximisée de la fonction de vraisemblance du modèle*

- ***Valeur*** : **Plus le BIC est bas, c**ela indique un meilleur compromis entre l'ajustement du modèle aux données et la complexité du modèle. Un BIC plus bas suggère que le modèle est préférable.

Résultat : BIC de -327006.46 indique que le modèle de clustering actuel semble être bien ajusté aux données.

- **Indice de *Calinski Harabasz* (CH) :** Évalue la dispersion entre les clusters par rapport à la dispersion à l'intérieur des clusters.
  - ***Valeur*** : Plus l'indice est élevé, meilleure est la séparation entre les clusters par rapport à la compacité intra-cluster.
  - ***Seuil de référence :*** Un indice de CH supérieur à 1000 est généralement considéré comme bon pour les grands ensembles de données.

  Résultat : Un indice de CH de 97 677,34, indiquant une excellente séparation entre les clusters et une compacité élevée à l'intérieur des clusters.
- **Analyse de variance (ANOVA) :** Le test F-value est de $97{,}677$ avec une très faible p-value ($< 2 \times 10^{-16}$), indiquant une différence significative entre les moyennes des clusters. Les résultats de l'ANOVA montrent qu'il y a des différences significatives dans la durée d'engagement restante en mois entre les clusters identifiés par cette méthode de clustering (*k-means*). Cela confirme que le clustering a réussi à identifier des groupes qui présentent des caractéristiques distinctes en termes de durée d'engagement restante.

Les résultats montrent une bonne qualité de clustering, avec des valeurs du coefficient de silhouette et de l'indice de *Calinski-Harabasz* suggérant une séparation inter-cluster adéquate et une cohésion intra-cluster robuste. Ces indicateurs confirment que les clusters obtenus par l'algorithme *K-means* sont bien définis, relativement homogènes à l'intérieur, et bien séparés les uns des autres.

Donc, ces résultats suggèrent que le clustering obtenu avec *K-means* est de bonne qualité, avec des clusters bien définis et peu de chevauchement entre eux.

Le graphique ci-dessous montre les taux de résiliation diminuent significativement avec l'augmentation de la durée d'engagement. Les contrats à long terme semblent offrir des avantages qui fidélisent les clients de manière plus efficace, tandis que les engagements à court terme ont le taux de résiliation le plus élevé, possiblement en raison de la flexibilité ou de la moindre perception d'engagement de ces contrats.

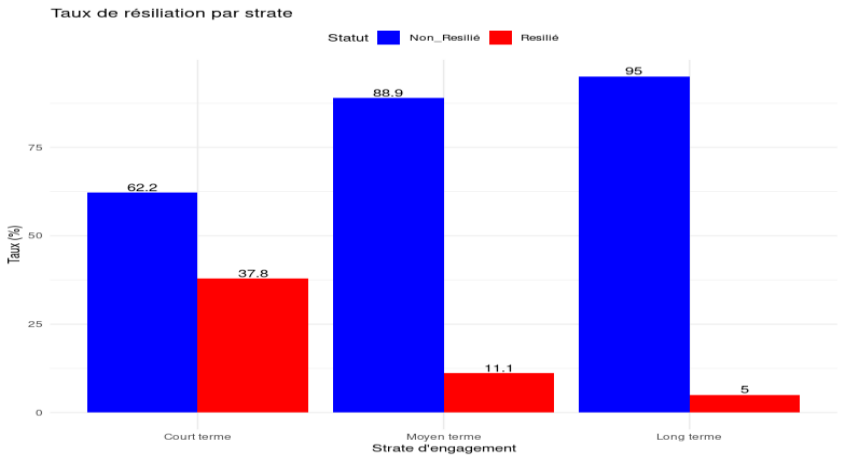


Figure 11 : Taux de résiliation par strate

Chapitre 3 Encodage des variables qualitatives

Pour notre étude visant à prédire la résiliation des clients, nous disposons d'une base de données contenant plusieurs variables catégorielles importantes. Ces variables comprennent des informations telles que le 'sexe', la 'csp' (catégorie socio-professionnelle), l''enseigne', le 'mode_paiement', le 'telephone_init', le 'telephone', la 'situation_impayes', et le 'segment'. Pour l'analyse initiale, nous avons choisi d'utiliser ces variables telles quelles, sans les encoder, afin d'intégrer les aspects socio-démographiques, comportementaux et transactionnels des clients dans notre modèle sans ajouter de complexité supplémentaire. Cependant, dans la suite de notre projet, nous explorerons également d'autres méthodes d'encodage pour ces variables, ce que nous décrirons plus en détail ultérieurement.

1 Sans encodage

1.1 Principe

Une variable qualitative (ou catégorielle) exprime des caractéristiques distinctes et non numériques, en contrastant avec les variables quantitatives qui sont mesurables et ordonnées numériquement. Les variables qualitatives peuvent être nominales, catégorisant les données sans ordre spécifique (par exemple, couleur des yeux, type de véhicule), ou ordinales, avec un ordre ou un rang entre les catégories sans que les différences soient uniformes (par exemple, niveau d'éducation, niveau de satisfaction). Elles jouent un rôle indisponsable dans l'analyse de données en permettant de segmenter et de catégoriser les données en groupes significatifs, ce qui est essentiel pour les analyses descriptives, les inférences statistiques et la modélisation prédictive. En R, un facteur est une structure de données utilisée pour représenter des variables catégorielles. Un facteur est un vecteur de valeurs discrètes, chaque valeur appartenant à un ensemble fini de niveaux (ou catégories). Les facteurs sont plus efficaces que les chaînes de caractères en termes de mémoire et sont optimisés pour les analyses statistiques (James, 2013).

1.2 Avantages et inconvénients

<u>Avantages</u> :

- ***Économie de Mémoire :*** Les facteurs utilisent des entiers internes pour représenter les niveaux des catégories, ce qui est plus efficace en termes de mémoire que les chaînes de caractères.
- ***Prévention des Erreurs :*** En définissant explicitement les niveaux possibles, les facteurs aident à prévenir les erreurs liées à l'usage incorrect ou à l'orthographe des catégories.

- ***Compatibilité avec les Méthodes Statistiques :*** De nombreuses fonctions statistiques et de modélisation en R sont conçues pour tirer parti des facteurs, traitant automatiquement les variables catégorielles de manière appropriée. Par exemple, les modèles de régression et de clustering utilisent des facteurs pour gérer correctement les variables catégorielles sans nécessiter de transformation supplémentaire.
- ***Clarté et Simplicité :*** En utilisant des facteurs, nous gardons les données dans leur forme la plus simple et la plus compréhensible, facilitant ainsi l'interprétation des résultats, notamment lors de la visualisation ou de l'analyse des clusters.
- ***Préservation de l'Information :*** En ne transformant pas ces variables en valeurs numériques arbitraires, nous évitons d'introduire des relations numériques non existantes entre les catégories. Cela permet de préserver l'intégrité de l'information originale et d'éviter de suggérer une relation d'ordre ou de distance inexistante entre les catégories.

Inconvénients :

- ***Limitation dans Certains Algorithmes :*** Certains algorithmes de machine learning nécessitent des données numériques et ne peuvent pas directement traiter des facteurs. Cela peut limiter le choix des algorithmes disponibles ou nécessiter des étapes de prétraitement supplémentaires.
- ***Scalabilité :*** Pour les très grands ensembles de données, l'utilisation de facteurs peut parfois poser des problèmes de scalabilité, en particulier lorsque le nombre de niveaux est très élevé.
- ***Analyse Avancée :*** Pour certaines analyses avancées nécessitant des mesures de similarité ou de distance, il peut être nécessaire de convertir les variables catégorielles en représentations numériques.

2 Encodage *One-Hot*

2.1 Principe

Le One-Hot Encoding est une technique utilisée pour convertir des variables catégorielles en une forme numérique que les algorithmes de machine learning peuvent utiliser directement. Cela implique la création de nouvelles colonnes binaires (0 ou 1) pour chaque niveau de la variable catégorielle. Chaque nouvelle colonne représente une catégorie unique, et pour chaque observation, la colonne correspondant à la catégorie présente prend la valeur 1, tandis que les autres colonnes prennent la valeur 0 *(RStudio Tutorials - "One-Hot Encoding in R")*. Supposons

que nous ayons une variable catégorielle "sexe" avec deux niveaux : "Homme" et "Femme". Voici comment le One-Hot Encoding fonctionne :

ID_client	Sexe
1	Homme
2	Femme
3	Homme
4	Femme

Après l'application du One-Hot Encoding, cela deviendra :

ID_client	Sexe_Homme	Sexe_Femme
1	1	0
2	0	1
3	1	0
4	0	1

2.2 Avantages et inconvénients

Avantages :

- ***Compatibilité avec les algorithmes de Machine Learning :*** De nombreux algorithmes de machine learning, tels que les régressions linéaires et les réseaux de neurones, ne peuvent pas travailler directement avec des variables catégorielles. Le One-Hot Encoding convertit ces variables en une forme qu'ils peuvent traiter.
- ***Préservation de l'Information :*** En utilisant des colonnes binaires distinctes, le One-Hot Encoding garantit qu'aucune information sur les catégories n'est perdue.
- ***Aucune Supposition d'Ordre :*** Contrairement à l'encodage ordinal, le One-Hot Encoding ne suppose pas de relation d'ordre entre les catégories, ce qui est idéal pour les variables nominales.

Inconvénients :

- ***La Malédiction de la Dimensionnalité :*** Si une variable catégorielle a un grand nombre de niveaux, le One-Hot Encoding peut entraîner une explosion du nombre de colonnes, augmentant ainsi la dimensionnalité du dataset. Cela peut rendre les modèles plus lents et moins efficaces.
- ***Sparité des Données :*** Le One-Hot Encoding produit des matrices éparses (avec beaucoup de 0), ce qui peut nécessiter plus de mémoire et de puissance de calcul.

- ***Problème de redondance :*** Lorsque toutes les catégories sont représentées par des colonnes binaires, une colonne peut être dérivée des autres, introduisant une redondance dans les données.

3 Encodage par fréquence

3.1 Principe

L'encodage de fréquence est une technique qui convertit les variables catégorielles en utilisant la fréquence d'apparition de chaque catégorie dans les données. Cela signifie que chaque catégorie est remplacée par le nombre d'occurrences de cette catégorie dans le dataset. C'est une méthode simple et efficace pour gérer les variables catégorielles, surtout lorsqu'il y a beaucoup de niveaux différents. Supposons que nous ayons une variable catégorielle "enseigne" avec les catégories suivantes : "A", "B", "C". Voici un exemple de comment l'encodage de fréquence fonctionne :

ID_client	Segment
1	A
2	B
3	A
4	C
5	B

La fréquence d'apparition des catégories est la suivante :

- "A" : 2 fois
- "B" : 2 fois
- "C" : 1 fois

Après l'application de l'encodage de fréquence, cela devient :

ID_client	Segment
1	2
2	2
3	2
4	1
5	2

3.2 Avantages et inconvénients

Avantages :

- ***Simplicité :*** Facile à implémenter et à comprendre.
- ***Réduction de la Dimensionnalité :*** Contrairement au One-Hot Encoding, l'encodage de fréquence ne génère pas de nouvelles colonnes, ce qui permet de garder la dimensionnalité du dataset inchangée.
- ***Incorporation d'Information Statistique :*** La fréquence des catégories peut parfois contenir des informations utiles pour les modèles de machine learning.

Inconvénients :

- ***Perte de l'Information de Catégorie :*** Ne conserve pas les informations sur les catégories uniques, ce qui peut être un problème si l'ordre ou la spécificité des catégories est important.
- ***Biais potentiel :*** Les catégories fréquentes peuvent dominer le modèle, introduisant un biais.

4 Encodage basé sur la cible

4.1 Principe

L'encodage basé sur la cible est une méthode où chaque catégorie est remplacée par la moyenne de la variable cible pour cette catégorie. Cette technique est particulièrement utile lorsque la variable cible est continue ou binaire, et peut améliorer les performances des modèles en introduisant des informations supplémentaires.

Supposons que nous ayons une variable catégorielle "segment" et une variable cible "flag_resiliation" indiquant si un client a résilié (1) ou non (0). Voici un exemple de comment l'encodage basé sur la cible fonctionne :

ID_client	Segment	Flag_Resiliation
1	A	1
2	B	0
3	A	1
4	C	0
5	B	0
6	A	0

La moyenne de "flag_resiliation" pour chaque catégorie "enseigne" est calculée :

- "A" : $\frac{(1+1+0)}{3} = 0{,}67$
- "B" : $\frac{(0+0)}{2} = 0$
- "C" : 0

Après l'application de l'encodage basé sur la cible, cela devient :

ID_client	Segment_Encoded
1	0.67
2	0
3	0.67
4	0
5	0
6	0.67

4.2 Avantage et inconvénients

Avantages :

- ***Utilisation de l'Information de la Cible :*** Introduit des informations supplémentaires directement liées à la variable cible.
- ***Réduction de la Dimensionnalité :*** Contrairement au One-Hot Encoding, cette méthode ne génère pas de nouvelles colonnes, ce qui maintient la dimensionnalité du dataset.
- ***Amélioration potentielle des performances :*** Peut améliorer les performances des modèles en intégrant des informations pertinentes.

Inconvénients :

- ***Fuite de Données :*** Risque de fuite d'informations si l'encodage est basé sur l'ensemble complet des données. Cela peut être atténué en utilisant des techniques comme le K-Fold Encoding.
- ***Biais Potentiel :*** Peut introduire un biais si certaines catégories sont sur-représentées ou sous-représentées.

Chapitre 4 Modélisation

Dans le cadre de notre projet d'analyse visant à prédire la résiliation des clients, notre objectif est de développer un modèle de machine learning capable d'identifier les clients à risque de résiliation. Pour ce faire, nous prévoyons de comparer les performances de deux modèles largement utilisés : le Random Forest et le GLM (Generalized Linear Model). Le Random Forest est réputé pour sa capacité à gérer des ensembles de données complexes comportant de nombreuses variables, tout en offrant une bonne précision prédictive. En revanche, le GLM se distingue par son interprétabilité accrue des résultats, ce qui permet de fournir des informations précieuses sur les facteurs influençant la résiliation des clients. Ces caractéristiques distinctes méritent d'être explorées afin de déterminer laquelle de ces approches convient le mieux à notre ensemble de données et à notre objectif de prédiction. En comparant ces deux modèles, nous visons à identifier celui qui est le plus performant et le plus approprié pour notre problématique spécifique de prédiction de la résiliation des clients.

1 Modèles linéaire généralisé GLM

1.1 Principe et utilisation du modèle

Le modèle linéaire généralisé (GLM) est une extension des modèles linéaires qui permet de modéliser des relations entre une variable dépendante et plusieurs variables indépendantes, même lorsque la variable dépendante suit une distribution non normale. Dans le contexte de classification binaire, comme la prédiction de la résiliation des clients, nous utilisons une régression logistique, qui est un type de GLM où la variable cible suit une distribution binomiale.

- Fonctionnement du GLM pour la Classification Binaire
  - ***Variable Dépendante :*** Dans notre cas, il s'agit de *flag_resiliation*, indiquant si un client a résilié (1) ou non (0).
  - ***Variables Indépendantes :*** Les variables explicatives ou prédictives, qui peuvent être continues ou catégorielles.

Le modèle GLM calcule la probabilité qu'un événement se produise (ici, la résiliation) en utilisant la fonction logit, qui est la log-transformation de la probabilité de l'événement :

$$logit(P) = \log\left(\frac{P}{1-P}\right) = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + \cdots + \beta_n x_n$$

1.2 Modèle GLM et Encodages Catégoriels

Le modèle linéaire généralisé (GLM) peut être utilisé avec différentes **méthodes d'encodage** pour les variables catégorielles, influençant ainsi son comportement et ses performances.

- Sans encodage, les variables catégorielles sont directement incluses dans le modèle sous forme de facteurs, avec R transformant automatiquement ces facteurs en variables indicatrices (dummy variables). Mathématiquement, chaque niveau de la variable catégorielle se voit attribuer son propre coefficient dans le modèle, représentant la différence par rapport à une catégorie de référence.
- Avec l'encodage one-hot, chaque niveau de la variable catégorielle est transformé en une variable binaire distincte. Ces variables binaires sont considérées comme des variables prédictives indépendantes, et le modèle attribue des coefficients à chaque colonne binaire pour estimer l'effet de chaque niveau sur la variable cible.
- L'encodage basé sur la fréquence remplace chaque catégorie par sa fréquence d'apparition relative dans le dataset. Le modèle utilise directement ces fréquences pour ajuster ses paramètres sans introduire de nouvelles variables, ce qui peut simplifier le modèle et être efficace dans certaines situations.
- L'encodage basé sur la cible remplace chaque catégorie par la moyenne de la variable cible pour cette catégorie. Cela permet au modèle de capturer les effets de chaque niveau sur la variable cible, fournissant ainsi une estimation directe de la probabilité de la variable cible en fonction des niveaux des variables catégorielles.

1.3 Comportement du Modèle

- ***Interprétation des Coefficients :*** Avec l'encodage one-hot, chaque coefficient représente l'effet spécifique d'un niveau par rapport à un niveau de référence. Avec les autres encodages, l'interprétation est plus directe, car les coefficients représentent directement les effets moyens sur la variable cible.
- ***Flexibilité du modèle :*** L'encodage one-hot peut augmenter la dimensionnalité et la complexité du modèle, tandis que les autres encodages maintiennent ou réduisent la dimensionnalité.
- ***Sensibilité aux Données :*** L'encodage basé sur la cible peut être sensible aux déséquilibres de classe et à la distribution des données. Les autres encodages sont moins sensibles à ces problèmes.
- ***Performances du Modèle :*** Les performances peuvent varier en fonction de la nature des données et de la relation entre les variables catégorielles et la variable cible. Il est

recommandé de tester plusieurs encodages et de sélectionner celui qui offre les meilleures performances.

2 Modèle de forêt aléatoire Random-Forest

2.1 Principe et utilisation du modèle

Le modèle Random Forest est un algorithme de machine learning robuste et flexible, bien adapté pour les tâches de classification et de régression. Il fonctionne en construisant une multitude d'arbres de décision lors de l'entraînement et en sortant la classe mode des classes individuelles des arbres pour la classification ou la moyenne des prédictions pour la régression (Hastie, T., 2009). L'un des principaux avantages du Random Forest est sa capacité à gérer des données de grande dimension et à capturer des interactions complexes entre les variables sans nécessiter un important pré-traitement des données (Breiman, L. (2001)).

2.2 Modèle Random Forest et Encodages Catégoriels

En ce qui concerne les variables catégorielles, le Random Forest peut les traiter de différentes manières selon le type d'encodage appliqué :

- Sans Encodage : Le Random Forest peut directement utiliser des variables catégorielles sans transformation, car de nombreux logiciels de machine learning, y compris les implémentations de Random Forest, peuvent gérer les variables catégorielles en interne. Les niveaux des variables catégorielles sont traités comme des valeurs distinctes dans les arbres de décision.
- Encodage One-Hot : Avec l'encodage one-hot, chaque niveau de la variable catégorielle est transformé en une colonne binaire distincte. Cela peut augmenter considérablement le nombre de variables dans le dataset, surtout lorsque les variables catégorielles ont de nombreux niveaux. Toutefois, le Random Forest est capable de gérer cette explosion dimensionnelle grâce à sa nature parcimonieuse, en sélectionnant seulement les variables les plus pertinentes pour chaque arbre.
- Encodage Basé sur la Fréquence : L'encodage basé sur la fréquence remplace chaque catégorie par sa fréquence d'apparition dans le dataset. Cela réduit le nombre de dimensions par rapport à l'encodage one-hot et permet au modèle d'intégrer directement des informations sur la distribution des catégories. Le Random Forest peut utiliser ces valeurs pour mieux différencier les catégories en fonction de leur fréquence.
- Encodage Basé sur la Cible : Cette méthode remplace chaque catégorie par la moyenne de la variable cible pour cette catégorie. Elle permet au modèle de capturer directement l'effet de chaque niveau sur la variable cible. Cependant, cette méthode peut introduire un biais si

elle n’est pas correctement régularisée, car le modèle peut apprendre des informations de la variable cible de manière trop directe.

En résumé, le Random Forest est très adaptable aux différents types d’encodage des variables catégorielles. Le choix de la méthode d’encodage peut avoir un impact sur la performance du modèle. Les encodages one-hot et basés sur la cible sont souvent utilisés pour leur capacité à capturer des informations détaillées sur les niveaux des variables catégorielles, tandis que les encodages sans transformation et basés sur la fréquence peuvent être plus simples et efficaces en termes de traitement et de performance computationnelle.

2.3 Comportement du Modèle Random Forest

- ***Interprétation des Variables :*** Contrairement aux modèles linéaires, Random Forest n’utilise pas de coefficients interprétables pour chaque variable. Il évalue plutôt l’importance des variables, ce qui indique quelles variables contribuent le plus aux prédictions. L’encodage one-hot dilue l’importance à travers plusieurs colonnes, tandis que d’autres encodages offrent une mesure plus directe de l’importance de chaque catégorie.
- ***Flexibilité du Modèle :*** Random Forest est très adaptable aux différents types d’encodage des variables catégorielles. L’encodage one-hot augmente la dimensionnalité mais est géré efficacement par le modèle grâce à l’agrégation sur de nombreux arbres. Les autres encodages, comme ceux basés sur la fréquence ou la cible, maintiennent ou réduisent la dimensionnalité, simplifiant ainsi le modèle.
- ***Sensibilité aux Données :*** Random Forest est généralement robuste aux déséquilibres de classe et aux distributions irrégulières des données. Cependant, l’encodage basé sur la cible peut introduire un biais si les classes sont très déséquilibrées, risquant de sur-apprendre sur les catégories minoritaires. L’encodage one-hot et l’encodage basé sur la fréquence sont moins susceptibles de provoquer ce problème.
- ***Performances du Modèle :*** Les performances de Random Forest peuvent être influencées par le choix de l’encodage des variables catégorielles. L’encodage one-hot capture bien les interactions complexes entre les catégories mais peut augmenter la complexité computationnelle. En revanche, les encodages basés sur la fréquence ou la cible simplifient la structure des données, favorisant ainsi l’apprentissage des relations directes avec la variable cible. Il est très important de tester plusieurs méthodes d’encodage pour déterminer celle qui optimise les performances spécifiques à chaque jeu de données.

Chapitre 5 Résultats et discussion

Dans ce chapitre, les résultats obtenus à partir des différentes méthodes de modélisation prédictive appliquées au jeu de données client sont analysés. L'objectif est de comprendre les facteurs influençant la résiliation des contrats et d'évaluer la performance des modèles utilisés pour prédire ce comportement. Les méthodes testées incluent le modèle GLM et les forêts aléatoires (Random Forest), chacune étant évaluée en termes d'aire sous la courbe ROC (AUC). L'impact de différentes techniques d'encodage des variables catégorielles sur la performance des modèles est également examiné, en comparant l'encodage one-hot, l'encodage basé sur la fréquence, et l'encodage basé sur la cible.

Les modèles prédictifs seront appliqués sur les données de 2023 pour anticiper les résiliations. Une attention particulière sera portée à la sélection des 2000 clients les plus susceptibles de résilier leur contrat, permettant ainsi de cibler des actions de fidélisation spécifiques. Ces analyses fourniront des insights pertinents pour améliorer la rétention client et optimiser les stratégies de fidélisation.

1 Encodage des variables catégorielles

Dans cette section, nous analysons la distribution des variables catégorielles de notre dataset en fonction du ***flag_resiliation***, qui indique si un client a résilié son contrat ou non. Les variables catégorielles étudiées incluent des caractéristiques socio-démographiques, comportementales et transactionnelles des clients : sexe, csp (catégorie socio-professionnelle), enseigne, mode_paiement, telephone_init, telephone, situation_impayes, et segment. L'objectif est de comprendre comment ces variables sont distribuées entre les clients ayant résilié et ceux n'ayant pas résilié, et d'identifier d'éventuelles tendances ou différences significatives.

Sexe : Indique le genre du client (Féminin ou Masculin).

CSP (Catégorie Socio-Professionnelle) : Représente la classification socio-professionnelle du client (Cadre, Commerçant, Employé, Étudiant, Fonctionnaire, Profession libérale, Sans emploi).

Enseigne : Indique le canal par lequel le client a souscrit à son contrat (Boutique, Grande distribution, Internet).

Mode de Paiement : Spécifie le mode de paiement utilisé par le client (Chèque, TIP, Virement).

Téléphone Initial : Catégorie du téléphone initial utilisé par le client (Bas de gamme, Carte SIM seule, Haut de gamme, Milieu de gamme).

Téléphone : Catégorie du téléphone actuel utilisé par le client (Bas de gamme, Haut de gamme, Milieu de gamme).

Situation Impayés : Indique la situation des impayés du client (A été en impayé, Aucun impayé, Est en impayé).

Segment : Classification du client dans un segment spécifique de marché (A, B, C)

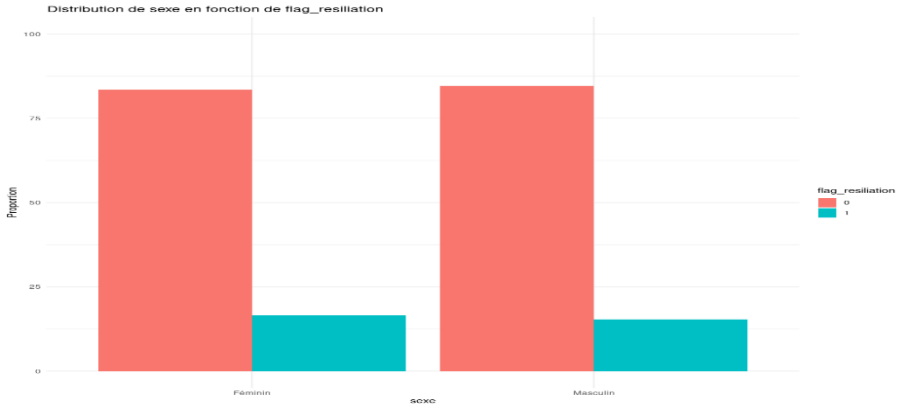


Figure 12 : Distribution de sexe en fonction de flag résiliation

Les clients féminins et masculins présentent des schémas similaires de rétention et d'attrition.

Pour les deux sexes, la majorité des clients, environ 80 %, ne sont pas partis.

Environ 20 % des clients des deux sexes sont partis.

Cela suggère que le sexe ne semble pas être un facteur significatif dans la décision de rester ou de partir, puisque les proportions de résilience sont presque identiques pour les femmes et les hommes.

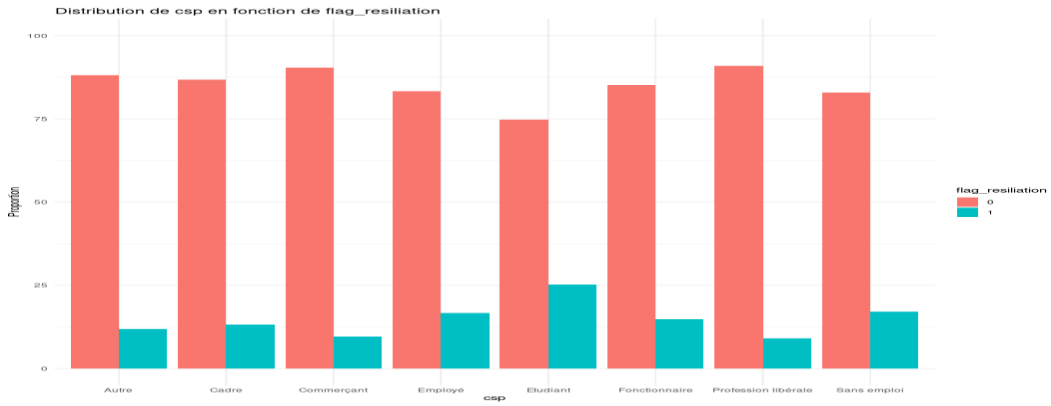


Figure 13 : Distribution de CSP par résiliation

Les catégories « Autre », « Commerçant », « Fonctionnaire » et « Profession libérale » ont un taux de rétention plus élevé, avec environ 80 % de clients qui ne partent pas.

Les catégories « Employé » et « Sans emploi » ont un taux de rétention légèrement inférieur, avec environ 70 % de clients ne partant pas.

Les catégories « Cadre » et « Etudiant » ont des taux de rétention d'environ 75 %, avec environ 25 % de départs.

Dans l'ensemble, le graphique indique que même si la majorité des clients de toutes les catégories du CSP ont tendance à rester, il existe des différences notables dans les proportions de ceux qui partent, les catégories « Employé » et « Sans emploi » ayant une plus forte tendance à partir.

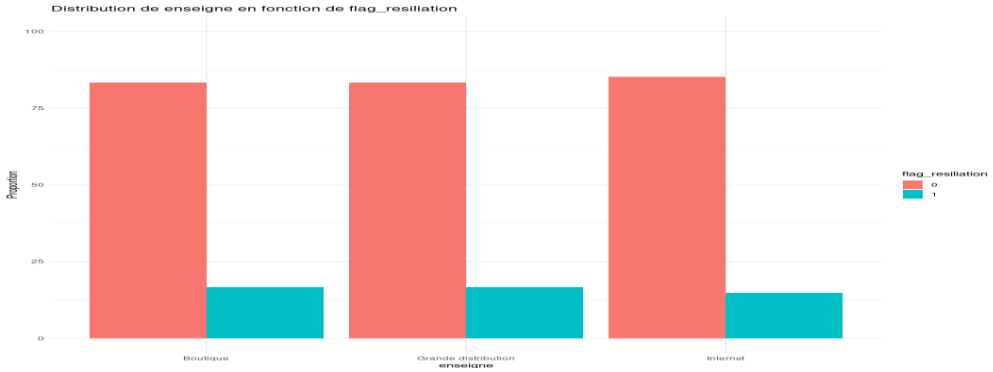


Figure 14 : Distribution de l'enseigne en fonction de la résiliation

La proportion de clients qui n'ont pas résilié est constante à environ 80% pour les trois canaux d'abonnement (Boutique, Grande distribution, Internet).

La proportion de clients qui ont résilié est également constante à environ 20% pour les trois canaux d'abonnement. Aucune des enseignes ne semble influencer de manière significative la décision de résiliation des clients. En conclusion, la décision de résiliation des clients semble indépendante de l'enseigne où ils se sont abonnés pour la première fois, car les proportions de résiliation sont similaires pour Boutique, Grande distribution et Internet.

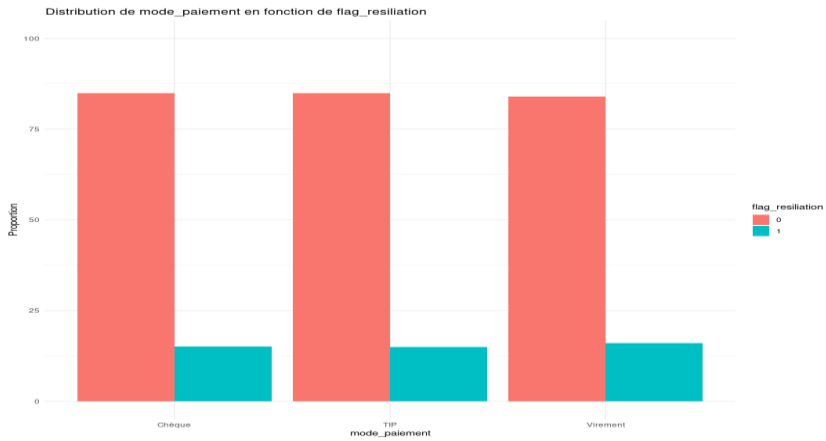


Figure 15 : Distribution du mode de paiement par résiliation

Le mode de paiement (Chèque, TIP, Virement) ne semble pas influencer significativement la probabilité de résiliation de l'abonnement téléphonique. Les proportions de résiliation sont relativement constantes à travers tous les modes de paiement, avec environ 20% de résiliations et 80% de non-résiliations.

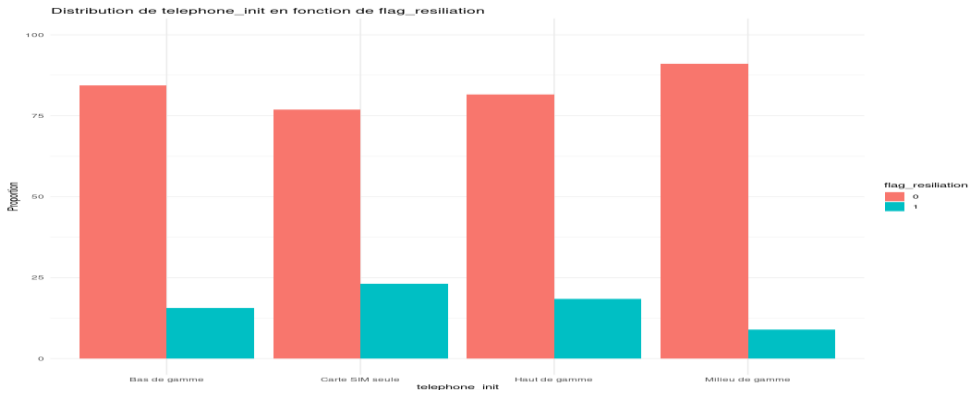


Figure 16 : Distribution de téléphone initial en fonction de la résiliation

Les clients ayant initialement une Carte SIM seule ont la proportion de résiliation la plus élevée parmi toutes les catégories de téléphones (environ 25%).

Les clients avec des téléphones Bas de gamme et Haut de gamme ont des proportions de résiliation similaires, autour de 20%.

Les clients avec des téléphones Milieu de gamme ont la proportion de résiliation la plus basse (environ 15%).

Le type de téléphone initial des clients semble avoir une influence sur la résiliation de l'abonnement téléphonique. Les clients ayant une Carte SIM seule sont plus susceptibles de résilier leur abonnement, tandis que ceux ayant des téléphones de milieu de gamme sont les moins susceptibles de le faire. Les téléphones bas et haut de gamme montrent des tendances intermédiaires avec des proportions de résiliation modérées.

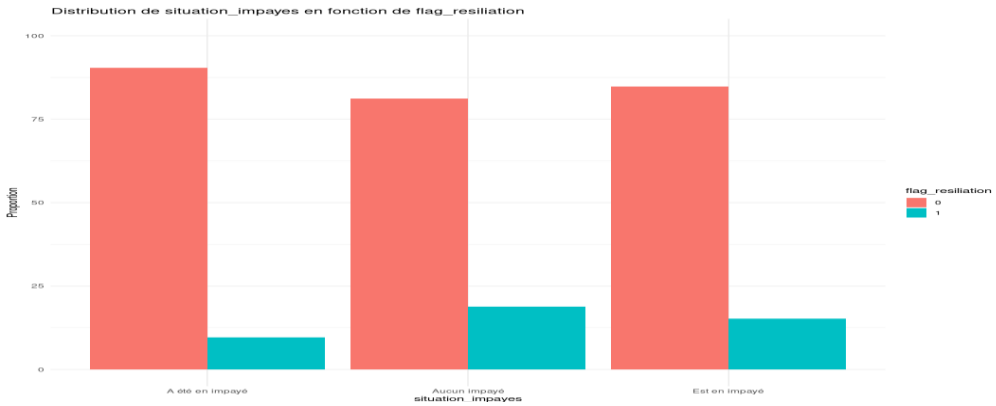


Figure 17 : Distribution de la situation des impayés par résiliation

La situation d'impayés des clients semble influencer la résiliation de l'abonnement téléphonique. Les clients qui ont actuellement des impayés ou qui n'ont jamais eu d'impayés ont des taux de résiliation plus élevés comparés à ceux qui ont été en impayé par le passé. Ces observations peuvent suggérer que les clients avec un historique d'impayés réglé ont une probabilité plus faible de résilier leur abonnement par rapport aux autres groupes.

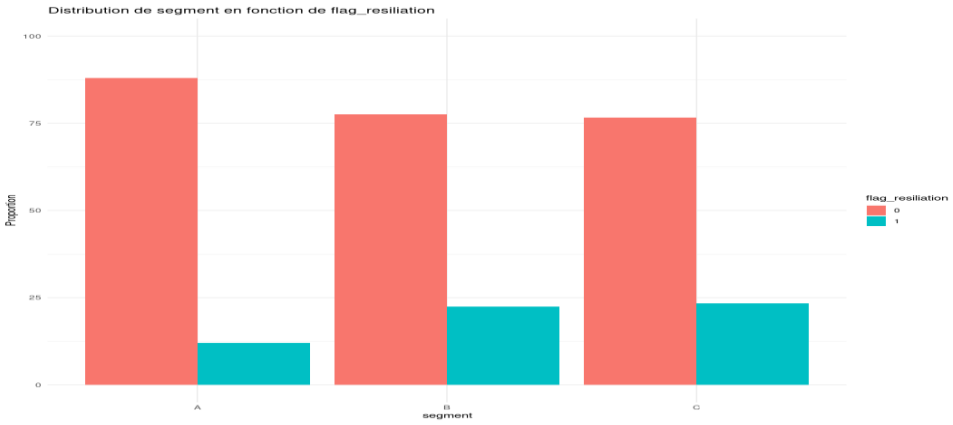


Figure 18 : Distribution de segment en fonction de la résiliation

Le graphique montre la distribution des segments des clients en fonction de leur statut de résiliation. Il est évident que dans le segment A, une majorité écrasante des clients n'ont pas résilié leur contrat (flag_resiliation = 0), représentant environ 90% des clients, tandis qu'environ 10% ont résilié. Pour le segment B, bien que la proportion de clients n'ayant pas résilié soit encore majoritaire, elle est moins marquée, avec environ 75% des clients n'ayant pas résilié et 25% ayant résilié. Une tendance similaire est observée dans le segment C, où environ 75% des clients n'ont pas résilié et 25% ont résilié.

Cette distribution indique que les clients du segment A sont beaucoup moins susceptibles de résilier leur contrat par rapport aux clients des segments B et C. Les segments B et C, ayant des proportions de résiliation comparables, montrent un comportement de résiliation similaire. Le segment A, avec son faible taux de résiliation, pourrait offrir des caractéristiques ou des avantages plus attractifs, incitant les clients à rester plus longtemps. Pour une compréhension approfondie, il serait pertinent d'explorer les facteurs spécifiques à ce segment, tels que la qualité du service, les offres spéciales, les programmes de fidélité, ou d'autres bénéfices distinctifs.

2 Préparation de données pour modélisation

La répartition des données en ensembles d'entraînement et de test est une étape cruciale dans la construction et l'évaluation de modèles prédictifs, tels que la régression logistique (GLM) et

les forêts aléatoires (Random Forest). Cette technique permet de mesurer la performance du modèle sur des données qu'il n'a pas vues pendant l'entraînement, ce qui aide à évaluer sa capacité de généralisation (Kuhn, M.,2013).

Dans le contexte de la modélisation, l'ensemble d'entraînement est utilisé pour ajuster les paramètres du modèle. Par exemple, dans une régression logistique, les coefficients du modèle sont ajustés pour minimiser une fonction de coût (comme l'erreur quadratique moyenne ou l'entropie croisée). De même, dans une forêt aléatoire, de nombreux arbres de décision sont construits sur des sous-ensembles aléatoires des données d'entraînement pour voter sur les prédictions finales.

L'ensemble de test, en revanche, est utilisé pour évaluer la performance du modèle. Après l'entraînement, les prédictions du modèle sont comparées aux vraies valeurs de l'ensemble de test pour calculer des métriques de performance comme l'accuracy, le F1-score, l'aire sous la courbe ROC (AUC), etc. Cela fournit une estimation plus réaliste de la performance du modèle sur des données nouvelles (Liaw, A., 2002). Voici comment on peut diviser les données en ensembles d'entraînement et de test en utilisant le package ***caret*** en R, en assurant une répartition de 75% pour l'entraînement et 25% pour le test :

Dans le code en annexe, ***createDataPartition*** de ***caret*** est utilisé pour créer une partition stratifiée des données, ce qui garantit que les proportions des classes cibles dans les ensembles d'entraînement et de test sont similaires à celles du jeu de données original. Cela est particulièrement important dans des cas de déséquilibre de classes.

2.1 Répartition du jeu d'Entraînement et de Test

Pour chaque jeu de données et pour chaque strate, un modèle de régression logistique est construit en utilisant les données d'entraînement, après avoir enlevé les colonnes non pertinentes (*id_client et strate_engagement*). Le modèle prédit ensuite les probabilités de résiliation sur les données de test, et sa performance est évaluée en calculant l'AUC (Area Under the ROC Curve), qui mesure la capacité du modèle à distinguer entre les classes (résiliation/non-résiliation). Un AUC élevé indique que le modèle a une bonne capacité de discrimination. Les AUC obtenus pour chaque méthode d'encodage sont comparés pour chaque strate, ce qui permet de déterminer quelle méthode d'encodage est la plus efficace pour prédire la résiliation dans chaque groupe d'engagement.

Dans cette section, nous présentons les dimensions des ensembles d'entraînement et de test pour chaque méthode d'encodage utilisée dans notre étude. Nous avons appliqué quatre méthodes différentes : sans encodage, *encodage one-hot*, encodage basé sur la fréquence, et

encodage basé sur la cible. Pour chacune de ces méthodes, les données ont été stratifiées en fonction de la variable *strate_engagement*, ce qui nous permet de comparer la taille des ensembles d'entraînement et de test pour chaque strate. Voici les dimensions des ensembles d'entraînement et de test pour chaque méthode d'encodage :

- ***Dimensions pour les Sans codage :***

Strate 1 (court terme) : $Train = 32002 \times 40$; $Test = 10176 \times 40$

Strate 2 (moyen terme) : $Train = 21164 \times 40$; $Test = 21014 \times 40$

Strate 3 (long terme) : $Train = 31190 \times 40$; $Test = 10988 \times 40$

- ***Dimensions pour la méthode One-hot encoding :***

Strate 1 (court terme) : $Train = 32002 \times 55$; $Test = 10176 \times 55$

Strate 2 (moyen terme) : $Train = 21164 \times 55$; $Test = 21014 \times 55$

Strate 3 (long terme) : $Train = 31190 \times 55$; $Test = 10988 \times 55$

- ***Dimensions pour la méthode Encodage basé sur la fréquence :***

Strate 1 (court terme) : $Train = 32002 \times 40$; $Test = 10176 \times 40$

Strate 2 (moyen terme) : $Train = 21164 \times 40$; $Test = 21014 \times 40$

Strate 3 (long terme) : $Train = 31190 \times 40$; $Test = 10988 \times 40$

- ***Dimensions pour la méthode Encodage basé sur la cible :***

Strate 1 (court terme) : $Train = 32002 \times 40$; $Test = 10176 \times 40$

Strate 2 (moyen terme) : $Train = 21164 \times 40$; $Test = 21014 \times 40$

Strate 3 (long terme) : $Train = 31190 \times 40$; $Test = 10988 \times 40$

Les dimensions des ensembles d'entraînement et de test varient en fonction de la méthode d'encodage utilisée. Les méthodes sans encodage, encodage basé sur la fréquence, et encodage basé sur la cible conservent le même nombre de variables (40), tandis que l'encodage one-hot augmente ce nombre à 55 en raison de sa nature qui crée des variables binaires pour chaque catégorie. Cela démontre l'impact de l'encodage des variables catégorielles sur la taille des matrices de données, ce qui peut influencer les performances des modèles d'apprentissage automatique. Dans les sections suivantes, nous examinerons les performances des modèles GLM entraînés avec chacune de ces méthodes d'encodage afin de comprendre comment ces différences dimensionnelles se traduisent en termes de précision des prédictions.

2.2 Résultats et Interprétation des Performances des Modèles GLM

Après avoir divisé nos données en ensembles d'entraînement et de test selon différentes méthodes d'encodage, nous avons entraîné des modèles GLM pour chaque strate d'engagement (court terme, moyen terme, long terme). Les performances des modèles ont été évaluées en

utilisant l'aire sous la courbe ROC (AUC) pour chaque méthode d'encodage. Les résultats obtenus montrent des variations dans les scores AUC en fonction de la méthode d'encodage et de la strate d'engagement. Ces résultats sont résumés dans le tableau ci-dessous :

Méthode d'encodage	Strate	AUC
Sans encodage	court terme	$AUC = 0.7584449$
	moyen terme	$AUC = 0.8030103$
	long terme	$AUC = 0.7747978$
One hot encoding	court terme	$AUC = 0.7584449$
	moyen terme	$AUC = 0.8030103$
	long terme	$AUC = 0.7747978$
Encodage par fréquence	court terme	$AUC = 0.7482115$
	moyen terme	$AUC = 0.7874976$
	long terme	$AUC = 0.7534581$
Encodage par cible	court terme	$AUC = 0.7524598$
	moyen terme	$AUC = 0.8020714$
	long terme	$AUC = 0.7731179$

En R, le modèle GLM (Generalized Linear Model) traite efficacement les variables catégorielles en utilisant des facteurs, en appliquant implicitement un codage de type "*dummy coding*" ou "*one-hot encoding*". Ce processus de codage des facteurs en variables binaires se réalise automatiquement en arrière-plan. Ainsi, le modèle GLM gère les facteurs de manière équivalente à un one-hot encoding explicite. Les résultats similaires entre l'utilisation des facteurs (variables qualitatives sans encodage) et le one-hot encoding s'expliquent par ce codage automatique des variables catégorielles dans le modèle GLM. Ces deux méthodes d'encodage semblent être les plus efficaces dans notre contexte, particulièrement pour les données de moyen terme, où le score AUC atteint 0.803.

L'encodage basé sur la fréquence montre des performances légèrement inférieures avec des scores AUC allant de 0.748 à 0.787. Cela peut être dû au fait que cette méthode ne capture pas aussi bien la variabilité des données comparée aux autres méthodes.

L'encodage basé sur la cible offre des performances intermédiaires avec des scores AUC allant de 0.752 à 0.802. Bien que cette méthode soit souvent utilisée pour sa capacité à capturer des relations non linéaires, dans notre cas, elle n'a pas surpassé les méthodes sans encodage et one-hot encoding.

En résumé, les résultats montrent que l'encodage des variables catégorielles a un impact significatif sur les performances des modèles GLM. Les méthodes sans encodage et one-hot encoding se sont avérées les plus performantes pour notre jeu de données. Ces résultats suggèrent que, pour des modèles de résiliation, il est crucial de choisir judicieusement la méthode d'encodage afin d'optimiser les performances prédictives.

2.3 Score stratifié du modèle GLM

Dans le cadre de l'évaluation des performances des modèles de régression logistique généralisée (GLM) sur des données stratifiées, il est crucial de synthétiser les résultats obtenus à partir des différentes strates pour obtenir une vision globale de l'efficacité des modèles. Pour ce faire, deux approches principales sont couramment utilisées : l***a moyenne simple des AUC (Area Under the Curve)*** et ***la moyenne pondérée des AUC***. Ces métriques permettent de quantifier la capacité prédictive des modèles sur l'ensemble des strates et facilitent ainsi la comparaison entre différentes méthodes d'encodage des données.

La moyenne simple des AUC calcule la valeur moyenne des AUC obtenues pour chaque strate, offrant ainsi une mesure globale de la performance des modèles. Cette approche est simple à mettre en œuvre et donne une vision équilibrée des performances sur l'ensemble des strates.

sans_encodage	encodage_one_hot	encodage_freq	encodage_target
0.7787510	0.7787510	0.7630557	0.7758830

En revanche, la moyenne pondérée des AUC permet d'attribuer des poids différents aux différentes strates en fonction de leur importance relative. Cette méthode peut être utilisée lorsque certaines strates sont considérées comme plus significatives que d'autres dans le contexte spécifique de l'analyse. Par exemple, des strates avec des volumes de données plus importants ou des strates représentant des segments de clients clés peuvent se voir attribuer des poids plus élevés pour refléter leur impact plus important sur l'ensemble des résultats.

sans_encodage	encodage_one_hot	encodage_freq	encodage_target
0.7783557	0.7783557	0.7620959	0.7756065

Ainsi, en utilisant ces deux approches, nous pouvons obtenir des scores de performance consolidés qui capturent efficacement l'efficacité des modèles GLM sur des données stratifiées, ce qui facilite la prise de décision dans le contexte de l'analyse des données et de la modélisation prédictive.

Les scores de propension calculés pour chaque méthode d'encodage reflètent la performance globale des modèles GLM sur l'ensemble des strates.

- Pour la méthode sans encodage et la méthode d'encodage one-hot, les scores de propension sont identiques, avec une valeur de 0.7787510. Cela indique que ces deux méthodes ont produit des performances similaires, en termes de capacité prédictive, sur l'ensemble des données stratifiées.

- La méthode d'encodage basée sur la fréquence a produit un score de propension légèrement inférieur, avec une valeur de 0.7630557. Cela suggère que l'utilisation de l'encodage basé sur la fréquence a entraîné des performances légèrement moins bonnes que les deux premières méthodes.
- Enfin, la méthode d'encodage basée sur la cible a produit un score de propension de 0.7758830, montrant des performances légèrement supérieures à celles de l'encodage basé sur la fréquence mais inférieures à celles de la méthode sans encodage et de l'encodage one-hot.

Lorsque les scores stratifiés sont pondérés en fonction de l'importance relative des strates, les tendances générales restent similaires, bien que les valeurs numériques puissent varier légèrement. Cela permet de prendre en compte l'impact différentiel des strates dans l'évaluation globale des performances des modèles.

2.4 Résultats et Interprétation des Performances du Modèle RandomForest

Nous présentons les résultats obtenus à partir de l'application du modèle Random Forest sur différentes strates temporelles (court terme, moyen terme et long terme) avec diverses méthodes d'encodage des variables catégorielles. Les performances des modèles sont évaluées à l'aide de l'AUC (Area Under the Curve).

Méthode d'encodage	Strate	AUC
Sans encodage	court terme	$AUC = 0.8004611$
	moyen terme	$AUC = 0.870543$
	long terme	$AUC = 0.8858352$
One hot encoding	court terme	$AUC = 0.7909125$
	moyen terme	$AUC = 0.8687572$
	long terme	$AUC = 0.8784437$
Encodage par fréquence	court terme	$AUC = 0.7934062$
	moyen terme	$AUC = 0.8659462$
	long terme	$AUC = 0.8803259$
Encodage par cible	court terme	$AUC = 0.7911878$
	moyen terme	$AUC = 0.8742857$
	long terme	$AUC = 0.8741718$

- Pour la strate court terme, le modèle sans encodage atteint la meilleure performance avec une AUC de 0.8005. Les autres méthodes d'encodage, bien que légèrement moins performantes, sont assez proches les unes des autres avec des AUC autour de 0.79.
- Pour la strate moyen terme, l'encodage par cible offre la meilleure performance avec une AUC de 0.8743, légèrement supérieure à celle obtenue sans encodage (0.8705). Les autres méthodes d'encodage fournissent des résultats comparables mais légèrement inférieurs.
- Pour la strate long terme, le modèle sans encodage obtient la meilleure performance avec une AUC de 0.8858. Les méthodes d'encodage par fréquence et one-hot suivent de près, tandis que l'encodage par cible est légèrement moins performant.

Les modèles sans encodage montrent de bonnes performances globales, notamment pour les strates 'moyen' et 'long' terme, avec des AUC respectives de **0.8705** et **0.8858**, ce qui suggère que les variables catégorielles contiennent suffisamment d'information pour être efficacement capturées par le modèle Random Forest. En revanche, l'encodage one-hot affiche des performances légèrement inférieures pour toutes les strates, probablement en raison de l'explosion du nombre de variables et de la perte d'information, ce qui est moins adapté aux Random Forest. L'encodage par fréquence, quant à lui, démontre des performances stables et compétitives, particulièrement utile dans des contextes de distributions catégorielles déséquilibrées. Enfin, l'encodage par cible présente des résultats variables et est le plus performant pour la strate moyen terme, ce qui suggère que l'utilisation de l'information de la variable cible peut améliorer les performances du modèle dans certains contextes, bien qu'il faille rester vigilant face au risque de surajustement.

2.5 Score stratifié du modèle Random Forest

Pour calculer ce score, nous pouvons utiliser une moyenne pondérée des AUC obtenues pour chaque méthode d'encodage sur les différentes strates. Une approche courante consiste à attribuer des poids égaux à chaque strate, mais il est possible d'ajuster les poids selon l'importance relative de chaque strate si nécessaire.

sans_encodage	encodage_one_hot	encodage_freq	encodage_target
0.8522798	0.8465594	0.8465594	0.8465484

Les différences de performance entre les modèles Random Forest avec et sans encodage one-hot peuvent s'expliquer par plusieurs facteurs. Tout d'abord, l'encodage one-hot transforme

chaque catégorie d'une variable en une nouvelle variable binaire, ce qui augmente considérablement le nombre de variables dans le jeu de données. Cette explosion du nombre de variables peut diluer l'importance des variables significatives et augmenter le bruit, rendant le modèle moins performant. De plus, les Random Forest sont des ensembles d'arbres de décision qui bénéficient souvent de la capacité à détecter des interactions entre variables. Lorsque les variables catégorielles sont représentées par un grand nombre de variables binaires, les interactions pertinentes peuvent devenir plus difficiles à capturer. En revanche, sans encodage ou avec des méthodes d'encodage qui conservent la structure d'information des variables catégorielles (comme l'encodage par fréquence ou par cible), les Random Forest peuvent plus facilement exploiter les informations pertinentes et les relations entre variables, ce qui conduit à une meilleure performance.

3 Comparaison des deux modèles

En comparant les deux modèles, le Random Forest se démarque comme le modèle le plus performant, particulièrement grâce à sa capacité à exploiter les variables catégorielles sans nécessiter d'encodage complexe. Cette efficacité est probablement due à la nature robuste et non paramétrique des Random Forest, qui peuvent gérer des variables catégorielles de manière intrinsèque sans perdre d'information. En revanche, le GLM, tout en étant performant, montre une sensibilité plus élevée aux méthodes d'encodage utilisées, ce qui peut limiter sa capacité prédictive dans des contextes variés. Ainsi, le Random Forest apparaît comme le modèle supérieur, offrant une flexibilité et une robustesse accrues dans le traitement des données catégorielles, ce qui le rend particulièrement adapté pour des analyses nécessitant une forte capacité de généralisation.

3.1 Score de propension

Le score de propension est une probabilité estimée qu'un individu présente un certain comportement ou appartienne à un groupe particulier, en fonction des variables observées. Dans cette étude, il quantifie la probabilité qu'un client résilie son contrat, en tenant compte de ses caractéristiques. Le score de propension a été utilisé pour évaluer les probabilités de résiliation des clients selon différents encodages des variables catégorielles. Il permet de comparer les performances des modèles et de déterminer les méthodes d'encodage les plus efficaces pour prédire la résiliation. Bien que puissant, son utilisation requiert une attention particulière à la sélection et à l'inclusion des variables pertinentes afin d'assurer des résultats fiables. La formule mathématique pour calculer le score de propension dans un modèle de régression logistique est la suivante :

$$Score\ de\ propension = P(Y = 1|X) = \frac{1}{1 - \exp\left[-\left(\beta_0 + \beta_1 X_1 + \beta_2 X_2 + \cdots + \beta_p X_p\right)\right]}$$

Où :

- $P(Y = 1|X)$ est la probabilité de résiliation (score de propension) ;
- β_0 est l'ordonnée à l'origine (intercept) ;
- $\beta_0,\ \beta_1,\ \beta_2, \ldots, \beta_p$ sont les coefficients de régression associés aux variables $X_1,\ X_2, \ldots, X_p$.

Intérêt et Avantages :

- ***Comparabilité :*** Le score de propension permet de comparer directement les probabilités de résiliation entre différents clients, indépendamment de leurs caractéristiques individuelles.
- ***Contrôle des Variables Confondantes :*** En ajustant pour les variables observées, le score de propension aide à contrôler les variables confondantes et à obtenir des estimations plus précises de l'effet des variables d'intérêt.
- ***Simplicité d'Interprétation :*** Le score de propension, étant une probabilité, est facile à interpréter et à communiquer à des parties prenantes non techniques.
- ***Utilisation dans les Études Observatoires :*** Il est particulièrement utile dans les études observationnelles où les groupes de traitement et de contrôle peuvent être très différents au départ.

Inconvénients :

- ***Dépendance aux Variables Observées :*** Le score de propension ne contrôle que pour les variables observées et incluses dans le modèle. Toute variable non observée ou omise peut biaiser les résultats.
- ***Nécessite un Modèle Bien Spécifié :*** La précision du score de propension dépend fortement de la spécification correcte du modèle de régression. Une mauvaise spécification peut entraîner des prédictions incorrectes.
- ***Complexité du Modèle :*** Plus il y a de variables et d'interactions, plus le modèle devient complexe à estimer et interpréter.

Règles d'interprétation des scores de propension :

- ***Interprétation des valeurs :***
    - ***Score proche de 0 :*** Indique une faible probabilité que l'événement se produise (par exemple, résiliation du contrat).
    - ***Score proche de 1 :*** Indique une forte probabilité que l'événement se produise.

- ***Utilisation pour la prise de décision :***
    - Les scores élevés peuvent être utilisés pour cibler des interventions spécifiques (comme des offres de rétention pour les clients à haut risque de résiliation).
    - Les scores bas peuvent indiquer des individus qui sont peu susceptibles de changer leur comportement ou de quitter un service.
- ***Comparaison des modèles :***
    - Utiliser les scores de propension pour évaluer et comparer les performances de différents modèles et méthodes d'encodage.
    - Un modèle avec des scores de propension bien calibrés fournit des prédictions plus fiables et précises.

En suivant ces règles d'interprétation, les scores de propension peuvent être utilisés efficacement pour la modélisation prédictive et la prise de décision stratégique. Les scores de propension permettent d'identifier les clients à haut risque de résiliation avec une certaine précision. Toutes les méthodes d'encodage fournissent des scores relativement cohérents, bien que certaines ajustent légèrement les probabilités prédites. Le choix de la méthode d'encodage doit être basé sur la performance et la spécificité des données.

```
> display_top5_propensity(combined_results_sans_encodage)
      flag_resiliation propensity_score
7211                 1        0.9092218
2480                 1        0.9068208
347                  1        0.8844126
26877                1        0.8798673
9036                 1        0.8752155
> cat("\nOne-hot encoding:\n")

One-hot encoding:
> display_top5_propensity(combined_results_one_hot)
      flag_resiliation propensity_score
7211                 1        0.9092218
2480                 1        0.9068208
347                  1        0.8844126
26877                1        0.8798673
9036                 1        0.8752155
> cat("\nEncodage basé sur la fréquence:\n")

Encodage basé sur la fréquence:
> display_top5_propensity(combined_results_freq)
      flag_resiliation propensity_score
20445                0        0.8729813
7211                 1        0.8643090
26877                1        0.8461635
1205                 0        0.8179640
9036                 1        0.8166983
> cat("\nEncodage basé sur la cible:\n")

Encodage basé sur la cible:
> display_top5_propensity(combined_results_target)
      flag_resiliation propensity_score
7211                 1        0.9058332
2480                 1        0.9021485
347                  1        0.9012438
9036                 1        0.8804056
20445                0        0.8732594
```

Pour le calcul d'une probabilité finale à partir des probabilités estimées sur plusieurs strates, le score de propension peut jouer un rôle clé en facilitant la combinaison des informations issues de chaque strate. Le score de propension, qui représente la probabilité estimée qu'un client

appartienne à une certaine classe (par exemple, le risque de résiliation), peut être utilisé comme un indicateur intermédiaire dans le processus de modélisation. Une approche courante consiste à calculer une moyenne pondérée des scores de propension obtenus pour chaque strate. Chaque strate peut être attribuée un poids basé sur son importance relative ou sur la qualité de ses prédictions.

- ***Moyenne pondérée des probabilités :***

Une méthode simple et efficace consiste à utiliser une moyenne pondérée des probabilités estimées pour chaque strate. Les poids peuvent être égaux ou ajustés en fonction de l'importance relative de chaque strate.

Formule :

$$Probabilité\ finale = \sum_{i=1}^{n} \omega_i p_i$$

Où : $\omega_i est\ le\ poids\ de\ la\ strate\ i;\ p_i\ est\ la\ probabilité\ estimée\ pour\ la\ strate\ i$

- ***Exemples d'application de ces méthodes :***

Supposons que nous ayons les probabilités estimées suivantes pour un client par différentes méthodes d'encodage et strates :

Méthode d'encodage	Court terme	Moyen terme	Long terme
Sans encodage	0.80	0.87	0.89
One-hot encoding	0.79	0.87	088

Si nous attribuons des poids égaux à chaque strate, la probabilité finale peut être calculée comme suit :

$$Probabilité\ final\ (sans\ encodage) = \frac{0.80 + 0.87 + 0.89}{3} = 0.8533$$

Cette méthode permet de synthétiser les informations temporelles variées en une seule probabilité globale, offrant ainsi une vue d'ensemble du risque de résiliation tout en exploitant les forces spécifiques de chaque modèle temporel.

4 Interprétation

La méthode de Shapley est un concept issu de la théorie des jeux coopératifs, adapté à l'interprétation des modèles de machine learning pour expliquer les prédictions individuelles.

Elle permet d'attribuer une valeur marginale à chaque caractéristique (ou méthode, dans le cas des méthodes d'encodage) en fonction de sa contribution à la prédiction du modèle.

❖ **Principes de la méthode de Shapley :**

Contexte : La méthode de Shapley est souvent utilisée pour expliquer comment chaque caractéristique (ou variable) contribue aux prédictions d'un modèle. Cela peut être particulièrement utile dans les modèles de machine learning où plusieurs variables sont utilisées pour faire des prédictions complexes.

- ***Concept clé :*** L'idée principale est d'attribuer une contribution marginale à chaque caractéristique en tenant compte de toutes les combinaisons possibles de caractéristiques.
- ***Calcul des valeurs de Shapley :*** Pour une caractéristique donnée X_i, les valeurs de Shapley sont calculées en prenant en compte toutes les permutations possibles dans lesquelles X_i peut-être ajouté à un sous-ensemble de caractéristiques S.
  - *Étape 1 :* Considérons toutes les permutations possibles de caractéristiques pour calculer les contributions marginales.
  - *Étape 2 :* Calculons la contribution moyenne de chaque caractéristique sur toutes les permutations.

Exemple concret :

Nous avons utilisé les méthodes d'encodage suivantes pour entraîner notre modèle de prédiction de résiliation des clients : Sans encodage ; Encodage one-hot ; Encodage par fréquence ; Encodage par cible.

Nous avons également stratifié nos données en trois catégories basées sur la durée d'engagement des clients : court terme, moyen terme et long terme.

Pour chaque combinaison de méthode d'encodage et de strate, nous avons entraîné un modèle de Random Forest et calculé les valeurs de Shapley pour expliquer les prédictions du modèle sur les données de test.

Une fois les valeurs de Shapley sont calculées pour chaque méthode d'encodage et chaque strate, ces valeurs permettent de :

- ***Comprendre l'importance des caractéristiques :*** Les valeurs de Shapley montrent comment chaque caractéristique contribue aux prédictions du modèle. Par exemple, pour une méthode d'encodage donnée et une strate spécifique (par exemple, long terme), nous pouvons vérifier

si CSP, ou d'autres caractéristiques ont une contribution plus significative à la prédiction de la résiliation des clients.

- ***Comparer les méthodes d'encodage :*** En comparant les valeurs de Shapley entre différentes méthodes d'encodage pour une même strate, nous déterminons quelle méthode d'encodage capture le mieux les relations entre les caractéristiques et la variable cible.
- ***Optimiser les stratégies de prédiction :*** En identifiant les caractéristiques les plus importantes pour chaque méthode d'encodage et chaque strate, nous optimisons les stratégies de prédiction de résiliation des clients en ajustant les variables clés.

❖ **<u>Modèle GLM pour moyen terme</u> :**

L'interprétation des informations fournies par l'explainer créé avec DALEX pour le modèle GLM (Generalized Linear Model) est la suivante :

```
> explainer <- explain(glm_model,
+                      data = test_data[, !names(test_data) %in% c("id_client","flag_resiliation", "propensity_score")],
+                      y = test_data$flag_resiliation,
+                      label = "GLM Model")
Preparation of a new explainer is initiated
  -> model label       :  GLM Model
  -> data              :  10544  rows  38  cols
  -> target variable   :  10544  values
  -> predict function  :  yhat.glm  will be used (  default  )
  -> predicted values  :  No value for predict function target column. (  default  )
  -> model_info        :  package stats , ver. 4.2.1 , task classification (  default  )
  -> predicted values  :  numerical, min =  4.790637e-06 , mean =  0.1601154 , max =  0.9779266
  -> residual function :  difference between y and yhat (  default  )
  -> residuals         :  numerical, min =  -0.926188 , mean =  0.003295096 , max =  0.9963316
  A new explainer has been created!
```

- ***Model label (étiquette du modèle) :*** GLM Model ; Cela indique que le modèle utilisé est un modèle linéaire généralisé.
- ***Data :*** 10544 lignes, 38 colonnes ; le jeu de données contient 10544 observations et 38 variables en entrée.
- ***Target variable (variable cible) :*** 10544 valeurs ; Il y a 10544 valeurs dans la variable cible.
- ***Predict function (fonction de prédiction) :*** yhat.glm sera utilisé (par défaut) ; La fonction de prédiction par défaut utilisée est `yhat.glm`, qui est associée aux prédictions du modèle GLM.
- ***Predicted values (valeurs prédites) :*** Les valeurs prédites vont de 4.790637e-06 (environ 0) à 0.9779266. Cela représente la plage des valeurs prédites par ce modèle pour la variable cible.
- ***Model info (informations sur le modèle) :*** package stats, ver. 4.2.1, task classification (par défaut) ; Le modèle utilisé est basé sur le package `stats` de R, version 4.2.1, et il est utilisé pour une tâche de classification.

- ***Residual function (fonction de résidus) :*** différence entre y et yhat (par défaut) ; La fonction de résidus par défaut utilisée est la différence entre les valeurs observées (`y`) et les valeurs prédites (`yhat`).
- ***Residuals (résidus) :*** Les résidus calculés vont de -0.926188 (valeur minimale) à 0.9963316 (valeur maximale), avec une moyenne de 0.003295096. Les résidus représentent les écarts entre les valeurs observées et les valeurs prédites par le modèle.

Les prédictions du modèle oscillent entre des valeurs proches de zéro et proches de 1. Les résidus moyens sont très faibles, indiquant que le modèle ajuste généralement bien les données, avec de faibles écarts moyens entre les prédictions et les valeurs réelles. Ces informations sont essentielles pour comprendre le comportement et la performance du modèle dans le cadre de problème de prédiction de résiliation (`flag_resiliation`).

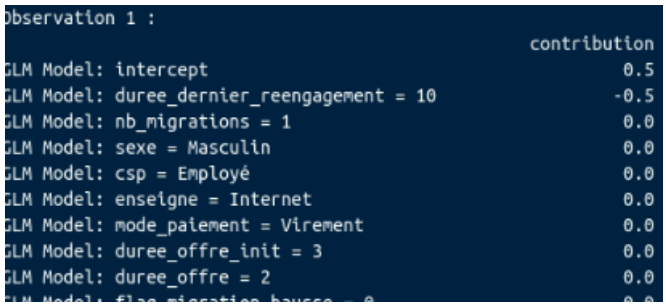
❖ **Modèle Random Forest pour moyen terme :**

```
> explainer <- explain(rf_model,
+                      data = test_data[, !names(test_data) %in% c("id_client", "flag_resiliation", "propensity_score")],
+                      y = test_data$flag_resiliation,
+                      label = "Random Forest Model")
Preparation of a new explainer is initiated
  -> model label       :  Random Forest Model
  -> data              :  10544  rows  38  cols
  -> target variable   :  10544  values
  -> predict function  :  yhat.randomForest  will be used (  default  )
  -> predicted values  :  No value for predict function target column. (  default  )
  -> model_info        :  package randomForest , ver. 4.7.1.1 , task regression (  default  )
  -> predicted values  :  numerical, min =  -1.504297e-15 , mean =  0.1628236 , max =  0.9511667
  -> residual function :  difference between y and yhat (  default  )
  -> residuals         :  numerical, min =  -0.9249667 , mean =  0.0005868677 , max =  0.9976
  A new explainer has been created!
```

La sortie de la préparation d'un nouvel explainer pour un modèle de forêt aléatoire (Random Forest Model). Les données utilisées contiennent 10 544 lignes et 38 colonnes, avec "flag_resiliation" comme variable cible. La fonction de prédiction par défaut yhat.randomForest est utilisée, mais aucune valeur prédite n'est indiquée pour la colonne cible. Les valeurs prédites sont numériques, avec un minimum de -1.5043, une moyenne de 0.1628, et un maximum de 0.9512. Les résidus, calculés comme la différence entre les valeurs réelles et les valeurs prédites, sont également numériques, avec un minimum de -0.9249, une moyenne de 0.0005807, et un maximum de 0.9976. Un nouvel explainer a été créé avec succès.

Les deux modèles présentent des valeurs prédites et des résidus proches, mais le modèle de forêt aléatoire montre une plus grande variabilité dans les valeurs prédites et des résidus moyens légèrement plus faibles, suggérant une performance légèrement meilleure dans l'ajustement des données.

❖ **Exemple de valeur de Shapley**



Observation 1 :

	contribution
GLM Model: intercept	0.5
GLM Model: duree_dernier_reengagement = 10	-0.5
GLM Model: nb_migrations = 1	0.0
GLM Model: sexe = Masculin	0.0
GLM Model: csp = Employé	0.0
GLM Model: enseigne = Internet	0.0
GLM Model: mode_paiement = Virement	0.0
GLM Model: duree_offre_init = 3	0.0
GLM Model: duree_offre = 2	0.0

Les résultats de Shapley pour une observation donnée montrent l'importance relative de chaque variable d'entrée dans la prédiction du modèle pour cette observation particulière. Les valeurs de contribution indiquent l'impact de chaque variable sur la prédiction.

Interprétation des résultats de Shapley pour une observation spécifique :

Prenons les résultats de l'observation 1 comme exemple (Annexe page 68): Pour l'observation 1, les valeurs de Shapley révèlent que le nombre de SMS envoyés en M4 (308) a la plus forte contribution positive (0.678) à la prédiction, suivi par la situation d'impayé (0.073) et une ancienneté de 1 an (0.078). À l'inverse, un nombre élevé de SMS envoyés en M5 (321) (-0.253) et en M3 (280) (-0.139), ainsi que la possession d'un téléphone de milieu de gamme au début (-0.115) et un nombre élevé de SMS envoyés en M1 (275) (-0.152) ont des contributions négatives significatives. La plupart des autres variables, telles que la durée du dernier réengagement, le volume d'appels et certaines caractéristiques démographiques, ont des contributions proches de zéro, indiquant qu'elles influencent peu la prédiction. Les résultats de Shapley montrent non seulement l'importance de chaque variable dans la prédiction, mais aussi comment le modèle utilise ces variables pour faire des prévisions. Cela permet d'identifier quelles variables sont les plus déterminantes et comment elles influencent la décision finale du modèle.

Cet exemple simplifié démontre l'importance de certaines variables spécifiques dans le modèle, mais il est crucial de vérifier chaque variable et son importance dans un contexte plus large et de répéter cette analyse pour un nombre significatif d'observations afin de tirer des conclusions solides. Ainsi, même si cet exemple est simplifié, il montre clairement comment les valeurs de Shapley peuvent être utilisées pour interpréter les prédictions d'un modèle de machine learning et comprendre les mécanismes sous-jacents de ces prédictions.

Conclusion et Perspectives

Le présent mémoire a exploré les techniques de modélisation prédictive appliquées à la problématique de la résiliation des clients dans le secteur des télécommunications. En utilisant diverses méthodes telles que la stratification des données et l'encodage des variables catégorielles, j'ai pu déployer des modèles robustes et performants.

Les modèles GLM et Random Forest ont été mis en œuvre et comparés, démontrant chacun des forces et des faiblesses spécifiques en fonction des caractéristiques des données et des objectifs de la prédiction. L'analyse des résultats a montré que la stratification des données et le choix judicieux des méthodes d'encodage sont cruciaux pour améliorer la précision et la généralisation des modèles prédictifs. Les modèles développés ont permis de mieux comprendre les facteurs influençant la résiliation des clients, fournissant ainsi des insights précieux pour la mise en place de stratégies de fidélisation plus efficaces.

Pour améliorer davantage les performances des modèles prédictifs et répondre aux défis posés par l'hétérogénéité des données, plusieurs pistes peuvent être envisagées :

- ***Combinaison de plusieurs modèles :*** La création d'un ensemble de modèles (ensemble learning) peut aider à tirer parti des forces de différents algorithmes. Par exemple, la combinaison des modèles GLM et Random Forest pourrait améliorer la précision globale en capturant à la fois les relations linéaires et non linéaires des données.
- ***Ajout de nouveaux types de modèles :*** Intégrer des algorithmes supplémentaires tels que XGBoost, CatBoost, et K-Nearest Neighbors (KNN) pourrait offrir des perspectives nouvelles et complémentaires. XGBoost et CatBoost sont particulièrement efficaces pour gérer les données catégorielles et les relations complexes, tandis que KNN peut apporter une approche différente basée sur la similarité des observations.
- ***Optimisation des hyperparamètres :*** Une optimisation systématique des hyperparamètres pour chaque modèle pourrait améliorer significativement les performances prédictives. Des techniques comme la recherche en grille (grid search) ou l'optimisation bayésienne peuvent être utilisées à cette fin.
- ***Utilisation de techniques de traitement avancées :*** L'application de techniques avancées de traitement des données, telles que l'ingénierie des caractéristiques (feature engineering) et l'analyse de sentiments sur les données textuelles, pourrait enrichir les modèles et améliorer leur précision (*Outils étudiés dans l'UE Web Minning*).

En suivant ces pistes, il est possible de développer des modèles encore plus robustes et performants, capables de mieux répondre aux défis posés par la complexité et la diversité des données dans le secteur des télécommunications.

Ce mémoire de fin d'études m'a offert l'opportunité d'approfondir les applications métier dans le domaine de l'économétrie, en particulier dans le secteur des télécommunications, à travers la modélisation et la mise en œuvre de de divers modèles de Machine Learning. Ce travail a été une occasion précieuse pour moi de mener des recherches approfondies, de synthétiser des informations complexes et de renforcer de manière significative mes connaissances en Machine Learning. J'ai ainsi pu acquérir une compréhension plus fine des techniques de modélisation prédictive et de leur application pratique, tout en développant des compétences précieuses dans ce domaine dynamique et en perpétuelle évolution.

Annexe : Code utilisé pour ce projet

- **Stratification des données**

Charger les données

```
base_telecom_2022_12 <- read.csv("base_telecom_2022_12.txt", header=TRUE, sep=";")
```

Stratification

```
# Sélection des variables pertinentes
base_telecom_filtered <- base_telecom_2022_12[is.finite(base_telecom_2022_12$duree
_engagement_restante_mois), ]
data_for_clustering <- base_telecom_filtered$duree_engagement_restante_mois
# Choix du nombre de clusters (ex. méthode du coude)
set.seed(151286)  # Pour la reproductibilité
wss <- sapply(2:5, function(k) kmeans(data_for_clustering, k)$tot.withinss)
plot(2:5, wss, type="b", pch = 19, frame = FALSE, xlab="Nombre de clusters", ylab=
"Total Within Sum of Squares")
# Charger les bibliothèques nécessaires
# Sélection des variables pertinentes
base_telecom_filtered <- base_telecom_2022_12[is.finite(base_telecom_2022_12$duree
_engagement_restante_mois), ]
data_for_clustering <- base_telecom_filtered$duree_engagement_restante_mois

data_with_target <- data.frame(duree_engagement_restante_mois = data_for_clusterin
g, flag_resiliation = base_telecom_filtered$flag_resiliation)
num_clusters <- 3
clusters_by_target <- split(data_with_target, data_with_target$flag_resiliation)
# Initialiser les résultats du clustering
kmeans_results <- list()
# Appliquer K-means sur chaque sous-groupe
for (target in names(clusters_by_target)) {
  subset_data <- clusters_by_target[[target]] $duree_engagement_restante_mois
  kmeans_results[[target]] <- kmeans(subset_data, centers = num_clusters)
  clusters_by_target[[target]] $cluster <- kmeans_results[[target]] $cluster
}

# Combiner les résultats des sous-groupes
final_clusters <- do.call(rbind, clusters_by_target)
```

- **Encodage des variables qualitatives**

Sans encodage

```
# Convertir les variables catégorielles en facteurs
variables_categorielles <- c("sexe","csp", "enseigne", "mode_paiement", "t
elephone_init", "telephone", "situation_impayes","segment")

df_data[, variables_categorielles] <- lapply(df_data[, variables_categorie
lles], as.factor)
print(head(df_data,5))
```

```
##         id_client flag_resiliation     sexe         csp            ense
igne
## 1 ID_200530279381                0  Féminin Sans emploi            Inte
rnet
## 2 ID_338394942292                0  Féminin     Employé Grande distribu
tion
## 3 ID_733906194071                0 Masculin    Etudiant            Inte
rnet
## 4 ID_559263815386                0  Féminin    Etudiant Grande distribu
tion
## 5 ID_263748365484                0  Féminin       Autre Grande distribu
tion
##   mode_paiement duree_offre_init duree_offre nb_migrations
## 1      Virement                4           8             1
## 2      Virement                1           1             0
## 3      Virement                3           3             1
## 4      Virement                4           2             1
## 5      Virement                4           2             1
##   flag_migration_hausse flag_migration_baisse nb_services
## 1                     1                     0           1
## 2                     0                     0           0
## 3                     0                     0           2
## 4                     0                     1           2
## 5                     0                     1           1
##   flag_personnalisation_repondeur flag_telechargement_sonnerie  telepho
ne_init
## 1                               0                           0 Milieu d
e gamme
## 2                               0                           0 Carte SI
M seule
## 3                               0                           1 Milieu d
e gamme
## 4                               0                           0    Bas d
e gamme
## 5                               0                           0 Milieu d
e gamme
##         telephone nb_reengagements situation_impayes vol_appels_m6
## 1 Milieu de gamme                1      Aucun impayé         40404
## 2   Haut de gamme                0      Aucun impayé         20875
## 3 Milieu de gamme                2   A été en impayé         22079
## 4   Haut de gamme                0      Aucun impayé         16586
## 5 Milieu de gamme                1      Aucun impayé          5487
##   vol_appels_m5 vol_appels_m4 vol_appels_m3 vol_appels_m2 vol_appels_m1
## 1         39438         37781         40136         42205         39144
## 2         19396         21062         20981         20680         20051
## 3         24814         31362         25557         28364         22830
## 4         15501         16105         15576         16658         17305
## 5          5002          4627          5111          5102          5071
##   flag_appels_vers_international flag_appels_depuis_international
## 1                              0                                 0
## 2                              0                                 0
## 3                              0                                 0
## 4                              0                                 0
## 5                              0                                 0
```

```
##   flag_appels_numeros_speciaux nb_sms_m6 nb_sms_m5 nb_sms_m4 nb_sms_m3
## 1                            0       142       137       130       139
## 2                            1        98        94       100       100
## 3                            0        67        72        85        74
## 4                            0       143       131       135       128
## 5                            0         6         4         2         5
##   nb_sms_m2 nb_sms_m1 segment strate_engagement age anciennete
## 1       146       133       A       Court terme  58          4
## 2       100        99       A       Court terme  24          1
## 3        79        68       A       Court terme  28          4
## 4       137       142       B       Court terme  27          3
## 5         6         6       B       Court terme  30          4
##   duree_engagement_restante anciennete_dernier_reengagement
## 1                         3                         15.77002
## 2                         0                         18.18770
## 3                         3                         15.14579
## 4                         6                         18.18770
## 5                         0                         12.45175
##   duree_dernier_reengagement
## 1                   15.00000
## 2                   17.70073
## 3                   15.00000
## 4                   17.70073
## 5                   12.00000
```

One-hot Encoding

```
# Encoder la variable 'sexe' manuellement en spécifiant les niveaux
encoded_sexe <- model.matrix(~ 0 + sexe, data = df_data)
# Supprimer la variable d'origine 'sexe' du dataframe
df_data_on_hot <- cbind(df_data, encoded_sexe)
df_data_on_hot  <- df_data_on_hot [, !names(df_data_on_hot ) %in% c("sexe"
)]
# Encoder les variables catégorielles en utilisant le codage one-hot
variables_categorielles_one_hot <- variables_categorielles[!variables_cate
gorielles %in% "sexe"]
encoded_matrix <- model.matrix(~ 0 + ., data = df_data_on_hot [, variables
_categorielles_one_hot])

# Créer un nouveau data frame avec les variables encodées
encoded_data_one_hot <- cbind(df_data_on_hot [, -which(names(df_data_on_ho
t ) %in% variables_categorielles_one_hot)], encoded_matrix)

# Afficher la structure de la nouvelle base de données encodée
str(encoded_data_one_hot)

## 'data.frame':    42178 obs. of  55 variables:
##  $ id_client                     : chr  "ID_200530279381" "ID_3383949
42292" "ID_733906194071" "ID_559263815386" ...
##  $ flag_resiliation              : Factor w/ 2 levels "0","1": 1 1 1
1 1 2 1 1 1 1 ...
##  $ duree_offre_init              : num  4 1 3 4 4 3 3 3 2 4 ...
##  $ duree_offre                   : num  8 1 3 2 2 2 3 6 2 4 ...
##  $ nb_migrations                 : int  1 0 1 1 1 1 3 5 0 3 ...
##  $ flag_migration_hausse         : Factor w/ 2 levels "0","1": 2 1 1
```

```
1 1 1 2 2 1 2 ...
##  $ flag_migration_baisse           : Factor w/ 2 levels "0","1": 1 1 1
2 2 2 2 2 1 2 ...
##  $ nb_services                     : int  1 0 2 2 1 2 1 5 4 2 ...
##  $ flag_personnalisation_repondeur : Factor w/ 2 levels "0","1": 1 1 1
1 1 1 2 1 1 1 ...
##  $ flag_telechargement_sonnerie    : Factor w/ 2 levels "0","1": 1 1 2
1 1 1 1 1 1 2 ...
##  $ nb_reengagements                : int  1 0 2 0 1 0 0 0 0 2 ...
##  $ vol_appels_m6                   : int  40404 20875 22079 16586 5487
12696 21155 21378 12366 612 ...
##  $ vol_appels_m5                   : int  39438 19396 24814 15501 5002
11766 20613 22187 13091 612 ...
##  $ vol_appels_m4                   : int  37781 21062 31362 16105 4627
13058 21292 25727 12131 612 ...
##  $ vol_appels_m3                   : int  40136 20981 25557 15576 5111
12490 20498 21188 13395 612 ...
##  $ vol_appels_m2                   : int  42205 20680 28364 16658 5102
11169 19228 21798 12938 612 ...
##  $ vol_appels_m1                   : int  39144 20051 22830 17305 5071
11232 19907 22556 13063 612 ...
##  $ flag_appels_vers_international  : Factor w/ 2 levels "0","1": 1 1 1
1 1 1 1 2 1 2 ...
##  $ flag_appels_depuis_international: Factor w/ 2 levels "0","1": 1 1 1
1 1 1 1 1 1 1 ...
##  $ flag_appels_numeros_speciaux    : Factor w/ 2 levels "0","1": 1 2 1
1 1 2 1 2 2 2 ...
##  $ nb_sms_m6                       : int  142 98 67 143 6 112 102 2 80
110 ...
##  $ nb_sms_m5                       : int  137 94 72 131 4 109 99 3 83 1
20 ...
##  $ nb_sms_m4                       : int  130 100 85 135 2 117 100 4 76
106 ...
##  $ nb_sms_m3                       : num  139 100 74 128 5 116 97 3 81
130 ...
##  $ nb_sms_m2                       : int  146 100 79 137 6 112 92 4 78
107 ...
##  $ nb_sms_m1                       : int  133 99 68 142 6 114 93 5 77 1
25 ...
##  $ strate_engagement               : Factor w/ 3 levels "Court terme",.
.: 1 1 1 1 1 1 1 1 1 1 ...
##  $ age                             : num  58 24 28 27 30 21 48 34 25 23
...
##  $ anciennete                      : num  4 1 4 3 4 1 2 2 1 3 ...
##  $ duree_engagement_restante       : num  3 0 3 6 0 4 12 12 7 2 ...
##  $ anciennete_dernier_reengagement : num  15.8 18.2 15.1 18.2 12.5 ...
##  $ duree_dernier_reengagement      : num  15 17.7 15 17.7 12 ...
##  $ sexeFéminin                     : num  1 1 0 1 1 1 1 1 1 0 ...
##  $ sexeMasculin                    : num  0 0 1 0 0 0 0 0 0 1 ...
##  $ cspAutre                        : num  0 0 0 0 1 0 0 0 0 0 ...
##  $ cspCadre                        : num  0 0 0 0 0 0 0 0 0 0 ...
##  $ cspCommerçant                   : num  0 0 0 0 0 0 0 0 0 0 ...
##  $ cspEmployé                      : num  0 1 0 0 0 1 0 0 1 0 ...
##  $ cspEtudiant                     : num  0 0 1 1 0 0 0 0 0 1 ...
```

```
##  $ cspFonctionnaire                : num  0 0 0 0 0 0 0 0 0 0 ...
##  $ cspProfession libérale          : num  0 0 0 0 0 0 0 0 0 0 ...
##  $ cspSans emploi                  : num  1 0 0 0 0 0 1 1 0 0 ...
##  $ enseigneGrande distribution     : num  0 1 0 1 1 0 1 1 0 1 ...
##  $ enseigneInternet                : num  1 0 1 0 0 0 0 0 1 0 ...
##  $ mode_paiementTIP                : num  0 0 0 0 0 0 0 0 0 0 ...
##  $ mode_paiementVirement           : num  1 1 1 1 1 1 0 1 1 1 ...
##  $ telephone_initCarte SIM seule   : num  0 1 0 0 0 0 1 0 0 0 ...
##  $ telephone_initHaut de gamme     : num  0 0 0 0 0 0 0 0 1 0 ...
##  $ telephone_initMilieu de gamme   : num  1 0 1 0 1 1 0 0 0 1 ...
##  $ telephoneHaut de gamme          : num  0 1 0 1 0 0 0 1 1 1 ...
##  $ telephoneMilieu de gamme        : num  1 0 1 0 1 1 1 0 0 0 ...
##  $ situation_impayesAucun impayé   : num  1 1 0 1 1 1 0 0 1 1 ...
##  $ situation_impayesEst en impayé  : num  0 0 0 0 0 0 1 0 0 0 ...
##  $ segmentB                        : num  0 0 0 1 1 1 0 0 1 0 ...
##  $ segmentC                        : num  0 0 0 0 0 0 0 0 0 0 ...

print(head(encoded_data_one_hot,5))

##         id_client flag_resiliation duree_offre_init duree_offre nb_migr
ations
## 1 ID_200530279381                0               4           8
1
## 2 ID_338394942292                0               1           1
0
## 3 ID_733906194071                0               3           3
1
## 4 ID_559263815386                0               4           2
1
## 5 ID_263748365484                0               4           2
1
##   flag_migration_hausse flag_migration_baisse nb_services
## 1                     1                     0           1
## 2                     0                     0           0
## 3                     0                     0           2
## 4                     0                     1           2
## 5                     0                     1           1
##   flag_personnalisation_repondeur flag_telechargement_sonnerie nb_reeng
agements
## 1                               0                            0
1
## 2                               0                            0
0
## 3                               0                            1
2
## 4                               0                            0
0
## 5                               0                            0
1
##   vol_appels_m6 vol_appels_m5 vol_appels_m4 vol_appels_m3 vol_appels_m2
## 1         40404         39438         37781         40136         42205
## 2         20875         19396         21062         20981         20680
## 3         22079         24814         31362         25557         28364
## 4         16586         15501         16105         15576         16658
```

```
## 5          5487          5002          4627          5111          5102
##   vol_appels_m1 flag_appels_vers_international flag_appels_depuis_inter
national
## 1         39144                              0
0
## 2         20051                              0
0
## 3         22830                              0
0
## 4         17305                              0
0
## 5          5071                              0
0
##   flag_appels_numeros_speciaux nb_sms_m6 nb_sms_m5 nb_sms_m4 nb_sms_m3
## 1                            0       142       137       130       139
## 2                            1        98        94       100       100
## 3                            0        67        72        85        74
## 4                            0       143       131       135       128
## 5                            0         6         4         2         5
##   nb_sms_m2 nb_sms_m1 strate_engagement age anciennete
## 1       146       133       Court terme  58          4
## 2       100        99       Court terme  24          1
## 3        79        68       Court terme  28          4
## 4       137       142       Court terme  27          3
## 5         6         6       Court terme  30          4
##   duree_engagement_restante anciennete_dernier_reengagement
## 1                         3                        15.77002
## 2                         0                        18.18770
## 3                         3                        15.14579
## 4                         6                        18.18770
## 5                         0                        12.45175
##   duree_dernier_reengagement sexeFéminin sexeMasculin cspAutre cspCadre
## 1                   15.00000           1            0        0        0
## 2                   17.70073           1            0        0        0
## 3                   15.00000           0            1        0        0
## 4                   17.70073           1            0        0        0
## 5                   12.00000           1            0        1        0
##   cspCommerçant cspEmployé cspEtudiant cspFonctionnaire cspProfession l
ibérale
## 1             0          0           0                0
0
## 2             0          1           0                0
0
## 3             0          0           1                0
0
## 4             0          0           1                0
0
## 5             0          0           0                0
0
##   cspSans emploi enseigneGrande distribution enseigneInternet mode_paie
mentTIP
## 1              1                           0               1
0
## 2              0                           1               0
```

```
0
## 3              0                          0                1
0
## 4              0                          1                0
0
## 5              0                          1                0
0
##   mode_paiementVirement telephone_initCarte SIM seule
## 1                     1                             0
## 2                     1                             1
## 3                     1                             0
## 4                     1                             0
## 5                     1                             0
##   telephone_initHaut de gamme telephone_initMilieu de gamme
## 1                           0                             1
## 2                           0                             0
## 3                           0                             1
## 4                           0                             0
## 5                           0                             1
##   telephoneHaut de gamme telephoneMilieu de gamme situation_impayesAucu
n impayé
## 1                      0                       1
1
## 2                      1                       0
1
## 3                      0                       1
0
## 4                      1                       0
1
## 5                      0                       1
1
##   situation_impayesEst en impayé segmentB segmentC
## 1                              0        0        0
## 2                              0        0        0
## 3                              0        0        0
## 4                              0        1        0
## 5                              0        1        0
```

Encodage de fréquence

```
encoded_data_freq <- df_data

for (col in variables_categorielles) {
  freq_table <- table(df_data[[col]])
  encoded_data_freq[[col]] <- freq_table[as.character(df_data[[col]])]
}
print(head(encoded_data_freq,5))

##         id_client flag_resiliation  sexe   csp enseigne mode_paiement
## 1 ID_200530279381                0 20492  2622    16896         38253
## 2 ID_338394942292                0 20492 13902    16006         38253
## 3 ID_733906194071                0 21681  6325    16896         38253
## 4 ID_559263815386                0 20492  6325    16006         38253
## 5 ID_263748365484                0 20492  8921    16006         38253
##   duree_offre_init duree_offre nb_migrations flag_migration_hausse
```

```
## 1                4           8             1                     1
## 2                1           1             0                     0
## 3                3           3             1                     0
## 4                4           2             1                     0
## 5                4           2             1                     0
##   flag_migration_baisse nb_services flag_personnalisation_repondeur
## 1                     0           1                                0
## 2                     0           0                                0
## 3                     0           2                                0
## 4                     1           2                                0
## 5                     1           1                                0
##   flag_telechargement_sonnerie telephone_init telephone nb_reengagement
s
## 1                            0           9750     20044
1
## 2                            0           7377     16588
0
## 3                            1           9750     20044
2
## 4                            0          16722     16588
0
## 5                            0           9750     20044
1
##   situation_impayes vol_appels_m6 vol_appels_m5 vol_appels_m4 vol_appel
s_m3
## 1             24989         40404         39438         37781         4
0136
## 2             24989         20875         19396         21062         2
0981
## 3             10613         22079         24814         31362         2
5557
## 4             24989         16586         15501         16105         1
5576
## 5             24989          5487          5002          4627
5111
##   vol_appels_m2 vol_appels_m1 flag_appels_vers_international
## 1         42205         39144                              0
## 2         20680         20051                              0
## 3         28364         22830                              0
## 4         16658         17305                              0
## 5          5102          5071                              0
##   flag_appels_depuis_international flag_appels_numeros_speciaux nb_sms_
m6
## 1                                0                            0       1
42
## 2                                0                            1
98
## 3                                0                            0
67
## 4                                0                            0       1
43
## 5                                0                            0
6
##   nb_sms_m5 nb_sms_m4 nb_sms_m3 nb_sms_m2 nb_sms_m1 segment strate_enga
```

```
gement
## 1       137       130       139       146       133   26221       Court
terme
## 2        94       100       100       100        99   26221       Court
terme
## 3        72        85        74        79        68   26221       Court
terme
## 4       131       135       128       137       142   13830       Court
terme
## 5         4         2         5         6         6   13830       Court
terme
##   age anciennete duree_engagement_restante anciennete_dernier_reengagem
ent
## 1  58          4                         3                       15.77
002
## 2  24          1                         0                       18.18
770
## 3  28          4                         3                       15.14
579
## 4  27          3                         6                       18.18
770
## 5  30          4                         0                       12.45
175
##   duree_dernier_reengagement
## 1                   15.00000
## 2                   17.70073
## 3                   15.00000
## 4                   17.70073
## 5                   12.00000
```

Encodage basé sur la cible

```r
# Exclure la variable strate_engagement de la liste des variables catégori
elles
variables_categorielles <- setdiff(variables_categorielles, "strate_engage
ment")

# Calculer la moyenne de la variable cible pour chaque catégorie de chaque
variable catégorielle
df_data$flag_resiliation <- as.numeric(as.character(df_data$flag_resiliati
on))
target_encoding_tables <- lapply(variables_categorielles, function(col) {
  df_data %>%
    group_by(!!sym(col)) %>%
    summarize(target_mean = mean(flag_resiliation, na.rm = TRUE)) %>%
    mutate(!!paste0("encoded_", col) := target_mean) %>%
    select(!!sym(col), !!sym(paste0("encoded_", col)))
})

# Fusionner les tables d'encodage basées sur la cible pour toutes les vari
ables catégorielles
encoded_data_target <- df_data
for (i in seq_along(variables_categorielles)) {
  encoded_data_target <- left_join(encoded_data_target, target_encoding_ta
bles[[i]], by = variables_categorielles[i])
```

```
}

# Supprimer les colonnes originales
encoded_data_target <- encoded_data_target[, !names(encoded_data_target) %
in% variables_categorielles]

# Afficher les premières lignes du dataframe encodé
print(head(encoded_data_target, 5))
```

```
##         id_client flag_resiliation duree_offre_init duree_offre nb_migr
ations
## 1 ID_200530279381                0                4           8
1
## 2 ID_338394942292                0                1           1
0
## 3 ID_733906194071                0                3           3
1
## 4 ID_559263815386                0                4           2
1
## 5 ID_263748365484                0                4           2
1
##   flag_migration_hausse flag_migration_baisse nb_services
## 1                     1                     0           1
## 2                     0                     0           0
## 3                     0                     0           2
## 4                     0                     1           2
## 5                     0                     1           1
##   flag_personnalisation_repondeur flag_telechargement_sonnerie nb_reeng
agements
## 1                               0                            0
1
## 2                               0                            0
0
## 3                               0                            1
2
## 4                               0                            0
0
## 5                               0                            0
1
##   vol_appels_m6 vol_appels_m5 vol_appels_m4 vol_appels_m3 vol_appels_m2
## 1         40404         39438         37781         40136         42205
## 2         20875         19396         21062         20981         20680
## 3         22079         24814         31362         25557         28364
## 4         16586         15501         16105         15576         16658
## 5          5487          5002          4627          5111          5102
##   vol_appels_m1 flag_appels_vers_international flag_appels_depuis_inter
national
## 1         39144                               0
0
## 2         20051                               0
0
## 3         22830                               0
0
## 4         17305                               0
```

```
0
## 5          5071                              0
0
##   flag_appels_numeros_speciaux nb_sms_m6 nb_sms_m5 nb_sms_m4 nb_sms_m3
## 1                            0       142       137       130       139
## 2                            1        98        94       100       100
## 3                            0        67        72        85        74
## 4                            0       143       131       135       128
## 5                            0         6         4         2         5
##   nb_sms_m2 nb_sms_m1 strate_engagement age anciennete
## 1       146       133       Court terme  58          4
## 2       100        99       Court terme  24          1
## 3        79        68       Court terme  28          4
## 4       137       142       Court terme  27          3
## 5         6         6       Court terme  30          4
##   duree_engagement_restante anciennete_dernier_reengagement
## 1                         3                         15.77002
## 2                         0                         18.18770
## 3                         3                         15.14579
## 4                         6                         18.18770
## 5                         0                         12.45175
##   duree_dernier_reengagement encoded_sexe encoded_csp encoded_enseigne
## 1                   15.00000    0.1658208   0.1712433        0.1484967
## 2                   17.70073    0.1658208   0.1665228        0.1665625
## 3                   15.00000    0.1535446   0.2528063        0.1484967
## 4                   17.70073    0.1658208   0.2528063        0.1665625
## 5                   12.00000    0.1658208   0.1188208        0.1665625
##   encoded_mode_paiement encoded_telephone_init encoded_telephone
## 1             0.1604319              0.0894359        0.19122930
## 2             0.1604319              0.2304460        0.09645527
## 3             0.1604319              0.0894359        0.19122930
## 4             0.1604319              0.1565004        0.09645527
## 5             0.1604319              0.0894359        0.19122930
##   encoded_situation_impayes encoded_segment
## 1                 0.1886430       0.1195988
## 2                 0.1886430       0.1195988
## 3                 0.0955432       0.1195988
## 4                 0.1886430       0.2237889
## 5                 0.1886430       0.2237889
```

- **Modélisation et interprétation**

```
library(glmnet)
library(caret)
library(pROC)
library(Matrix)
library(doParallel)
library(foreach)
library(DALEX)
set.seed(151286)  # Pour la reproductibilité
```

```r
# réduire la taille pour générer rapidement le code
sample_size <- 1000
sampled_indices <- sample(nrow(df_data), size = sample_size)
df_data_sample <- df_data[sampled_indices, ]
encoded_data_one_hot_sample <- encoded_data_one_hot[sampled_indices, ]
encoded_data_freq_sample <- encoded_data_freq[sampled_indices, ]
encoded_data_target_sample <- encoded_data_target[sampled_indices, ]

# Diviser les données en 75% entraînement et 25% test
train_indices <- createDataPartition(df_data_sample$flag_resiliation, p =
0.75, list = FALSE)
train_data <- df_data_sample[train_indices, ]
test_data <- df_data_sample[-train_indices, ]

train_indices_one_hot <- createDataPartition(encoded_data_one_hot_sample$f
lag_resiliation, p = 0.75, list = FALSE)
train_data_one_hot <- encoded_data_one_hot_sample[train_indices_one_hot, ]
test_data_one_hot <- encoded_data_one_hot_sample[-train_indices_one_hot, ]

train_indices_freq <- createDataPartition(encoded_data_freq_sample$flag_re
siliation, p = 0.75, list = FALSE)
train_data_freq <- encoded_data_freq_sample[train_indices_freq, ]
test_indices_freq <- setdiff(seq_len(nrow(encoded_data_freq_sample)), trai
n_indices_freq)
test_data_freq <- encoded_data_freq_sample[test_indices_freq, ]

train_indices_target <- createDataPartition(encoded_data_target_sample$fla
g_resiliation, p = 0.75, list = FALSE)
train_data_target <- encoded_data_target_sample[train_indices_target, ]
test_data_target <- encoded_data_target_sample[-train_indices_target, ]
convert_to_numeric <- function(data) {
  data[] <- lapply(data, function(x) {
    if (is.factor(x)) {
      as.numeric(as.character(x))
    } else if (is.character(x)) {
      as.numeric(x)
```

```
    } else {
      x
    }
  })
  return(data)
}
train_data_one_hot <- convert_to_numeric(train_data_one_hot)
test_data_one_hot <- convert_to_numeric(test_data_one_hot)
sparse_train_data_one_hot <- as(as.matrix(train_data_one_hot[, -which(name
s(train_data_one_hot) %in% c("flag_resiliation", "id_client", "strate_enga
gement"))]), "dgCMatrix")
sparse_test_data_one_hot <- as(as.matrix(test_data_one_hot[, -which(names(
test_data_one_hot) %in% c("flag_resiliation", "id_client", "strate_engagem
ent"))]), "dgCMatrix")

# Fonction pour entraîner un modèle glmnet et retourner les données de tes
t avec les scores de propension
train_and_evaluate_glmnet <- function(train_data, test_data, sparse = FALS
E) {
  if (sparse) {
    y_train <- train_data_one_hot$flag_resiliation
    y_test <- test_data_one_hot$flag_resiliation
  } else {
    x_train <- as.matrix(train_data[, -which(names(train_data) == "id_clie
nt","flag_resiliation")])
    y_train <- train_data$flag_resiliation
    x_test <- as.matrix(test_data[, -which(names(test_data) == "id_client"
,"flag_resiliation")])
    y_test <- test_data$flag_resiliation
  }

  # Entraîner un modèle glmnet
  cv_fit <- cv.glmnet(x_train, y_train, family = "binomial")

  # Prédire les probabilités de résiliation sur les données de test
  predicted_probabilities <- predict(cv_fit, newx = x_test, s = "lambda.mi
```

```
n", type = "response")

  # Ajouter les scores de propension aux données de test
  test_data$propensity_score <- predicted_probabilities

  # Évaluer les performances en calculant le score de propension (AUC)
  auc <- roc(y_test, predicted_probabilities)$auc
  test_data$predicted_probabilities <- predicted_probabilities

  # Retourner les données de test avec les scores de propension et le scor
e AUC
  list(test_data_with_scores = test_data, auc = auc)
}

# Fonction pour entraîner et récupérer les données avec scores pour chaque
méthode
train_and_collect_glmnet <- function(stratified_train_test, sparse = FALSE
) {
  lapply(stratified_train_test, function(stratum_data) {
    train_data <- stratum_data$train
    test_data <- stratum_data$test
    result <- train_and_evaluate_glmnet(train_data, test_data, sparse)
    result$test_data_with_scores
  })
}

# Diviser les données en ensembles d'entraînement et de test pour chaque s
trate
stratified_train_test <- lapply(unique(df_data_sample$strate_engagement),
function(stratum) {
  train_indices <- which(df_data_sample$strate_engagement != stratum)
  test_indices <- which(df_data_sample$strate_engagement == stratum)
  list(train = df_data_sample[train_indices, ], test = df_data_sample[test
_indices, ])
})
```

```r
stratified_train_test_freq <- lapply(unique(encoded_data_freq_sample$strat
e_engagement), function(stratum) {
  train_indices <- which(encoded_data_freq_sample$strate_engagement != str
atum)
  test_indices <- which(encoded_data_freq_sample$strate_engagement == stra
tum)
  list(train = encoded_data_freq_sample[train_indices, ], test = encoded_d
ata_freq_sample[test_indices, ])
})

stratified_train_test_target <- lapply(unique(encoded_data_target_sample$s
trate_engagement), function(stratum) {
  train_indices <- which(encoded_data_target_sample$strate_engagement != s
tratum)
  test_indices <- which(encoded_data_target_sample$strate_engagement == st
ratum)
  list(train = encoded_data_target_sample[train_indices, ], test = encoded
_data_target_sample[test_indices, ])
})

# Récupérer les données de test avec scores pour chaque méthode
test_data_with_scores_sans_encodage <- train_and_collect_glmnet(stratified
_train_test)

test_data_with_scores_freq <- train_and_collect_glmnet(stratified_train_te
st_freq)

test_data_with_scores_target <- train_and_collect_glmnet(stratified_train_
test_target)

# Combiner les résultats en un seul data frame pour chaque méthode
combine_results <- function(test_data_with_scores_list) {
  do.call(rbind, test_data_with_scores_list)
}

combined_results_sans_encodage <- combine_results(test_data_with_scores_sa
ns_encodage)
combined_results_freq <- combine_results(test_data_with_scores_freq)
combined_results_target <- combine_results(test_data_with_scores_target)

# Fonction pour afficher les 5 premières lignes triées par score de propen
sion (décroissant)
```

```r
# et sélectionner uniquement les colonnes flag_resiliation et propensity_s
core
display_top5_propensity <- function(combined_results) {
  sorted_results <- combined_results[order(-combined_results$propensity_sc
ore), ]
  top5 <- head(sorted_results[, c("id_client","flag_resiliation", "propens
ity_score")], 5)
  print(top5)
}
train_data <- train_data[, !names(train_data) %in% c("id_client")]
test_data <- test_data[, !names(test_data) %in% c("id_client")]
# Boucle pour calculer les valeurs de Shapley pour plusieurs observations


glm_model <- glm(flag_resiliation ~ ., data = train_data, family = binomia
l)

explainer <- explain(glm_model,
                     data = test_data[, !names(test_data) %in% c("id_clien
t","flag_resiliation", "propensity_score")],
                     y = test_data$flag_resiliation,
                     label = "GLM Model")

## Preparation of a new explainer is initiated
##   -> model label       :  GLM Model
##   -> data              :  250  rows  38  cols
##   -> target variable   :  250  values
##   -> predict function  :  yhat.glm  will be used (  default  )
##   -> predicted values  :  No value for predict function target column.
(  default  )
##   -> model_info        :  package stats , ver. 4.3.3 , task classificat
ion (  default  )
##   -> predicted values  :  numerical, min =  0.0001356145 , mean =  0.17
52949 , max =  0.9091647
##   -> residual function :  difference between y and yhat (  default  )
##   -> residuals         :  numerical, min =  -0.9091647 , mean =  -0.023
2949 , max =  0.9872134
##   A new explainer has been created!

# Fonction pour calculer les valeurs de Shapley pour chaque observation da
ns un ensemble de données

calculate_shapley_values <- function(explainer, data) {
  shap_values_list <- list()
  for (i in 1:nrow(data)) {
    observation <- data[i, !names(data) %in% c("id_client","flag_resiliati
on", "propensity_score")]
    shap_values <- predict_parts(explainer, new_observation = observation)
    shap_values_list[[i]] <- shap_values
  }
  return(shap_values_list)
}

shapley_values <- calculate_shapley_values(explainer, test_data)
```

```r
# Print Shapley values for each observation
for (i in 1:length(shapley_values)) {
  cat("Observation", i, ":\n")
  print(shapley_values[[i]])
  cat("\n")
}
```

```
##                     Observation                       1                     :
##                                                                  contribution
## GLM Model: intercept                                                   0.175
## GLM Model: nb_sms_m4 = 308                                             0.678
## GLM Model: nb_sms_m6 = 309                                             0.017
## GLM Model: vol_appels_m5 = 35340                                       0.018
## GLM Model: duree_dernier_reengagement = 3                              0.013
## GLM Model: vol_appels_m2 = 32160                                       0.010
## GLM Model: vol_appels_m1 = 27600                                      -0.017
## GLM Model: vol_appels_m6 = 33230                                      -0.006
## GLM Model: anciennete_dernier_reengagement = 3.351                    -0.017
## GLM Model: vol_appels_m3 = 28520                                       0.006
## GLM Model: strate_engagement = Long terme                             -0.015
## GLM Model: nb_sms_m2 = 302                                            -0.017
## GLM Model: duree_offre_init = 6                                        0.017
## GLM Model: nb_sms_m3 = 280                                            -0.139
## GLM Model: nb_sms_m5 = 321                                            -0.253
## GLM Model: telephone_init = Milieu de gamme                           -0.115
## GLM Model: nb_sms_m1 = 275                                            -0.152
## GLM Model: situation_impayes = A été en impayé                         0.073
## GLM Model: duree_offre = 2                                             0.075
## GLM Model: duree_engagement_restante = 0                              -0.043
## GLM Model: anciennete = 1                                              0.078
## GLM Model: flag_migration_hausse = 0                                  -0.077
## GLM Model: telephone = Haut de gamme                                  -0.033
## GLM Model: sexe = Féminin                                              0.032
## GLM Model: vol_appels_m4 = 33020                                       0.018
## GLM Model: nb_migrations = 2                                           0.016
## GLM Model: flag_appels_vers_international = 1                         -0.024
## GLM Model: flag_telechargement_sonnerie = 0                            0.019
## GLM Model: enseigne = Grande distribution                             -0.024
```

```
## GLM Model: nb_services = 4                                    -0.019
## GLM Model: flag_migration_baisse = 1                           0.018
## GLM Model: nb_reengagements = 1                               -0.010
## GLM Model: flag_appels_numeros_speciaux = 1                    0.012
## GLM Model: csp = Employé                                       0.010
## GLM Model: flag_personnalisation_repondeur = 0                 0.011
## GLM Model: age = 41                                           -0.005
## GLM Model: mode_paiement = Virement                            0.004
## GLM Model: flag_appels_depuis_international = 0               -0.007
## GLM Model: segment = A                                        -0.005
## GLM Model: prediction
```

Bibliographie

1. Breiman, L. (2001), "Random Forests", *Machine Learning*, 45(1) : 5-32. DOI : 10.1023/A :1010933404324.
2. Kuhn, M., & Johnson, K. (2013), "Applied Predictive Modeling", *Springer*.
3. Hastie, T., Tibshirani, R., & Friedman, J. (2009), "The Elements of Statistical Learning : Data Mining, Inference, and Prediction", *Springer*.
4. Jain, A. K. (2010). Data clustering : 50 years beyond K-means. Pattern recognition letters, 31(8), 651-666.
5. Kaufman, L., & Rousseeuw, P. J. (2009). Finding groups in data : an introduction to cluster analysis. John Wiley & Sons.
6. Tibshirani, R., Walther, G., & Hastie, T. (2001). Estimating the number of clusters in a data set via the gap statistic. Journal of the Royal Statistical Society : Series B (Statistical Methodology), 63(2), 411-423.
7. RStudio Tutorials - "One-Hot Encoding in R" : Cette ressource fournit une explication détaillée du One-Hot Encoding en utilisant R, en couvrant les concepts de base ainsi que des exemples pratiques.
8. Hastie, T., Tibshirani, R., & Friedman, J. (2009). The Elements of Statistical Learning : Data Mining, Inference, and Prediction. Springer.
9. Breiman, L. (2001). Random Forests. Machine Learning, 45(1), 5-32.
10. Louppe, G. (2014). Understanding Random Forests : From Theory to Practice. arXiv preprint arXiv :1407.7502
11. Kuhn, M., & Johnson, K. (2013). *Applied Predictive Modeling*. Springer. Ce livre couvre en détail les méthodes de partitionnement des données et leur importance dans la modélisation prédictive.
12. Liaw, A., & Wiener, M. (2002). Classification and regression by RandomForest. *R News*, 2(3), 18-22. Cet article présente l'algorithme Random Forest et discute de son application dans le logiciel R.
13. Venables, W. N., & Ripley, B. D. (2002). Modern Applied Statistics with S. Springer.
14. James, G., Witten, D., Hastie, T., & Tibshirani, R. (2013). An Introduction to Statistical Learning : With Applications in R. Springer.
15. Grolemund, G., & Wickham, H. (2016). R for Data Science : Import, Tidy, Transform, Visualize, and Model Data. O'Reilly Media.