Agence nationale de la Statistique et de la Démographie
Ecole nationale de la Statistique et de l'Analyse économique Pierre Ndiaye

GROUPE DE TRAVAIL

Sous le thème :

Prédiction de la survenance de sinistre en assurance voyage : utilisation du machine learning

---

Présenté par :
M. GUEYE Ousseynou
M. TIENDREBEOGO Panongmanégré
Elève-ingénieur statisticien économiste

Sous la supervision de :

M. BANCE Youssouf - **Actuaire**

Année scolaire 2024-2025

Décharge

***

LES OPINIONS EMISES DANS CE DOCUMENT SONT PROPRES A SON AUTEUR ET NE SAURAIENT EN AUCUN CAS ENGAGER LA RESPONSABILITE DE L'AGENCE NATIONALE DE LA STATISTIQUE ET DE LA DEMOGRAPHIQUE (ANSD) OU DE L'ÉCOLE NATIONALE DE LA STATISTIQUE ET DE L'ANALYSE ÉCONOMIQUE PIERRE NDIAYE (ENSAE) DE DAKAR.

Avant-propos

***

Au terme de notre formation d'Ingénieur Statisticien Économiste (ISE) à l'École nationale de la Statistique et de l'Analyse économique Pierre Ndiaye (ENSAE), nous avons l'honneur de présenter ce mémoire de Groupe de Travail (GT), fruit d'une collaboration entre deux étudiants passionnés par l'application des méthodes statistiques avancées aux défis contemporains du secteur assurantiel.

Ce travail de recherche s'inscrit dans le cadre de notre dernière année de formation à l'ENSAE, établissement d'excellence rattaché à l'Agence nationale de la Statistique et de la Démographie (ANSD) et membre distingué du Réseau des Écoles de Statistiques Africaines (RESA). Notre parcours académique, riche en enseignements théoriques et pratiques, trouve son aboutissement dans cette étude qui vise à résoudre une problématique concrète en partenariat avec la startup LeaGa, spécialisée dans l'assurance voyage.

La prédiction de la survenance de sinistres en assurance voyage représente un défi majeur, notamment en raison du déséquilibre inhérent aux bases de données dans ce secteur. Notre projet explore comment les algorithmes d'équilibrage peuvent améliorer les performances des modèles de machine learning dans ce contexte particulier, contribuant ainsi à l'amélioration des pratiques actuarielles et à l'innovation dans le domaine de l'assurance.

Cette étude n'aurait pu être menée à bien sans le soutien précieux de nos encadrants académiques et professionnels. Nous tenons à exprimer notre profonde gratitude envers notre tuteur, Monsieur Youssouf BANCE, pour ses conseils avisés et sa disponibilité tout au long de ce parcours. Nos remerciements s'adressent également à l'ensemble du corps professoral de l'ENSAE qui, par la qualité de leur enseignement, nous a fourni les outils nécessaires pour mener à bien cette recherche.

Nous espérons que ce travail contribuera modestement à l'avancement des connaissances dans le domaine de l'actuariat et du machine learning appliqué à l'assurance, tout en témoignant de l'excellence de la formation dispensée à l'ENSAE.

Liste des sigles et abréviations

***

ML : machine Learning

ACP : Analyse en composante factorielle

IARD : Incendie, accidents et risques divers

SVM : machines à vecteurs de support

BVIP : Behavior-based Vehicle Insurance Pricing

KDD : Knowledge Discovery in Databases

DM : Data Mining

ASCA : Ant System-based Clustering Algorithm

AK : Ant-based K-means

SMOTE : Synthetic Minority Over-sampling Technique

ADASYN : Adaptive Synthetic Sampling

ROSE : Random Oversampling Examples

MWMOTE : Majority Weighted Minority Oversampling Technique

CNN : Condensed Nearest Neighbor

ENN : Edited Nearest Neighbor,

NCL : Neighborhood Cleaning Rule,

TL : Tomek Links,

OSS : One-Sided Selection,

CPM : K-medoids, Class Purity Maximization,

SBC : Under-Sampling Based on Clustering,

AHC : Agglomerative Hierarchical Clustering

SPIDER : Selective Pre-Processing of Imbalance Data

GB : Gradient Boosting

RF : Random Forest

KNN : K-Nearest Neighbors

RU : Random Under sampling

LACE : Length of stay, Acuity of admission, Comorbidities, Emergency department visits

AK : Ant-based K-means

Table des matières

***

Table des figures

***

Liste des tableaux

***

Résumé

***

L'assurance voyage, branche de l'IARD, affronte un défi majeur dans la prédiction des sinistres : le déséquilibre des classes. La grande majorité des contrats ne génère aucun sinistre, compliquant l'entraînement des modèles prédictifs qui tendent à favoriser la classe majoritaire. Cette étude vise à améliorer la détection de la classe minoritaire (sinistres) via deux approches complémentaires : les méthodes data-level (modification des données par rééchantillonnage) et algorithm-level (adaptation des algorithmes).

La méthodologie combine prétraitement des données, sélection de variables, et application de diverses techniques de rééchantillonnage (SMOTE, ADASYN, ROS, RUS, CNN, SMOTE+ENN, SMOTE+TOMEK) sur plusieurs modèles d'apprentissage (régression logistique, arbres de décision, Random Forest, XGBoost, réseaux neuronaux). Les performances sont évaluées par le G-Mean et l'AUC, métriques adaptées aux problèmes déséquilibrés.

Les résultats révèlent la supériorité du Random Forest avec des scores G-mean supérieurs à 21% pour la plupart des techniques de rééchantillonnage, notamment avec CNN (22%). La régression logistique (20-21%) et les réseaux neuronaux (17-19%) suivent, tandis que les arbres de décision simples et XGBoost offrent des performances inférieures. En termes d'AUC, Random Forest atteint 76% avec certaines techniques (ADASYN, SMOTE+ENN, SMOTE+TOMEK).

Une optimisation supplémentaire par ajustement du seuil de probabilité à 0.44 pour le Random Forest associé au CNN a permis d'atteindre un G-mean de 75.68%, avec 73.19% de vrais positifs et 21.74% de faux positifs. Cette combinaison constitue la solution la plus efficace pour améliorer la prédiction des sinistres en assurance voyage, offrant aux assureurs des outils concrets pour optimiser leur gestion des risques et personnaliser leurs offres.

Abstract

***

Travel insurance, a branch of property and casualty insurance, faces a major challenge in predicting claims : class imbalance. The vast majority of contracts generate no claims, complicating the training of predictive models that tend to favor the majority class. This study aims to improve the detection of the minority class (claims) through two complementary approaches : data-level methods (modifying data through resampling) and algorithm-level methods (adapting algorithms).

The methodology combines data preprocessing, variable selection, and application of various resampling techniques (SMOTE, ADASYN, ROS, RUS, CNN, SMOTE+ENN, SMOTE+TOMEK) on several learning models (logistic regression, decision trees, Random Forest, XGBoost, neural networks). Performance is evaluated using G-Mean and AUC, metrics adapted to imbalanced problems.

Results reveal the superiority of Random Forest with G-mean scores above 21% for most resampling techniques, particularly with CNN (22%). Logistic regression (20-21%) and neural networks (17-19%) follow, while simple decision trees and XGBoost offer lower performance. In terms of AUC, Random Forest reaches 76% with certain techniques (ADASYN, SMOTE+ENN, SMOTE+TOMEK).

Additional optimization by adjusting the probability threshold to 0.44 for Random Forest associated with CNN achieved a G-mean of 75.68%, with 73.19% true positives and 21.74% false positives. This combination constitutes the most effective solution for improving prediction of claims in travel insurance, offering insurers concrete tools to optimize their risk management and personalize their offerings.

Introduction

***

Le secteur de l'assurance constitue l'un des piliers majeurs de l'économie moderne, en apportant une protection financière contre une diversité de risques (santé, biens, vie, entreprises, etc.) et en contribuant à la stabilité économique globale. Parmi ses branches, l'assurance IARD (Incendie, Accidents et Risques Divers) inclut notamment l'assurance voyage, qui joue un rôle particulier en sécurisant les déplacements internationaux et en protégeant les voyageurs contre les imprévus tels que les annulations, les urgences médicales ou la perte de bagages. Avec l'évolution des comportements des consommateurs et l'essor des solutions numériques, l'assurance voyage connaît aujourd'hui une transformation rapide, marquée par l'intégration des technologies avancées et des services personnalisés.

Parmi ces innovations, l'apprentissage automatique (machine learning, ML) s'impose comme un levier clé. Alors que les méthodes traditionnelles de prévision et d'évaluation des risques reposaient sur des approches manuelles lentes et fortement dépendantes de l'expertise humaine, les algorithmes de ML permettent aujourd'hui de traiter de grandes masses de données, de capturer des relations complexes et de produire des prédictions précises (Di Franco & Santurro, 2021). L'accès à des volumes de données croissants, la baisse des coûts de stockage et l'augmentation des capacités de calcul ont considérablement renforcé le potentiel de ces approches (Schmidt et al., 2019).

Cependant, dans le domaine spécifique de l'assurance voyage, la prédiction des sinistres présente une difficulté majeure : le déséquilibre des classes. Les bases de données utilisées affichent en effet une répartition très asymétrique, où la grande majorité des contrats ne donne lieu à aucun sinistre (classe majoritaire), tandis que seuls quelques cas isolés enregistrent des sinistres (classe minoritaire). Or, les modèles classiques de ML ont tendance à privilégier la prédiction correcte de la classe majoritaire, au détriment de la minoritaire, ce qui réduit considérablement leur utilité pratique.

Face à ce constat, la problématique centrale que ce document cherche à explorer est la suivante :

> Comment améliorer la capacité des modèles de machine learning à prédire les sinistres, c'est-à-dire à mieux identifier la classe minoritaire, dans un contexte marqué par le déséquilibre des données ?

Répondre à cette question est essentiel, car des prédictions biaisées peuvent fausser la tarification, détériorer la gestion des risques et entraîner des pertes financières importantes. Plusieurs approches ont été développées pour remédier à ce problème, regroupées en deux grandes familles (Haixiang, L. Yijing et al., 2017) :

- les méthodes data-level (ou externes), qui ajustent l'équilibre des classes au niveau des données ;
- les méthodes algorithm-level (ou internes), qui modifient ou adaptent les algorithmes eux-mêmes afin qu'ils accordent plus d'attention à la classe minoritaire.

Des techniques comme l'analyse en composantes principales (ACP) peuvent par ailleurs être mobilisées pour améliorer la qualité des données en réduisant leur dimensionnalité et en optimisant les performances des modèles prédictifs.

L'objectif général de ce travail est de développer et évaluer des modèles prédictifs performants pour anticiper les sinistres en assurance voyage, malgré la forte asymétrie des données disponibles. Plus spécifiquement, il s'agit de :

- Présenter le cadre conceptuel et les spécificités de l'assurance IARD et, plus particulièrement, de l'assurance voyage.
- Analyser les principales approches de gestion du déséquilibre des classes dans les tâches de classification binaire.
- Mettre en œuvre une méthodologie combinant les approches data-level et algorithm-level.
- Évaluer la performance des modèles obtenus à l'aide de métriques adaptées et analyser les résultats.

Pour structurer cette démarche, le travail s'articulera autour du plan suivant :

- Présentation de l'assurance IARD et de l'assurance voyage ;
- Revue des principales approches de résolution du problème de déséquilibre des classes ;
- Méthodologie ;
- Résultats et analyses.

En apportant des solutions concrètes à ce défi méthodologique, ce travail vise à offrir aux gestionnaires et praticiens de l'assurance voyage de nouvelles perspectives pour améliorer la précision des prédictions, mieux maîtriser les risques financiers et optimiser les services offerts aux assurés.

Chapitre 1

Assurance IARD et le cas spécifique de l'assurance voyage

***

L'assurance IARD (Incendie, accidents et risques divers) ou assurance non-vie est une catégorie d'assurances qui offre une protection des biens matériels d'un assuré contre divers risques potentiels. Elle regroupe notamment les assurances auto, habitation, multirisques professionnelles. Sa spécificité est de couvrir les biens des assurés contre divers types de dommages tels que l'incendie, le vol, les dégâts des eaux, et autres risques divers. Ainsi, l'assurance voyage s'inscrit dans le cadre plus large de l'assurance IARD en tant que produit spécifique visant à protéger les biens et les personnes contre les risques liés aux voyages. Quant à l'assurances vie, elle qui vise à couvrir les individus contre des risques liés à leur santé ou leur vie.

1.1 Une brève historique

La notion d'assurance voyage découle d'une évolution historique, étroitement liée à l'expansion du commerce international. En effet, dès l'Antiquité, les Babyloniens par la pratique du "prêt à la grosse aventure", permettaient aux marchands de ne pas rembourser un prêt en cas de perte de cargaison. Les Grecs et Romains mirent en place des mécanismes similaires pour partager les risques liés au transport maritime. Au Moyen Âge, avec l'essor du commerce méditerranéen, l'assurance maritime se formalise davantage, et puis en 1435, l'Ordonnance de Barcelone, promulguée par Jacques Ier d'Aragon, fut le premier texte réglementant l'assurance voyage. Elle fut suivie par d'autres textes réunis dans le Consulat de la mer, codifiant les usages du commerce naval. Au XIX siècle, la révolution industrielle

et les nouveaux moyens de transport, comme le train, ont entraîné une diversification des assurances.

Sous une forme beaucoup plus moderne, l'assurance voyage émerge véritablement de nouveau à cette période avec une nouvelle dynamique, notamment grâce à l'essor du secteur touristique. Elle s'est structurée rapidement pour couvrir les risques spécifiques aux déplacements internationaux, des garanties comme l'annulation de voyage, l'assistance médicale ou la perte de bagages deviennent standards. Ainsi, les compagnies ont étendu leurs prestations afin de répondre aux exigences variées des voyageurs, et désormais, l'assurance voyage constitue un secteur consolidé, offrant des solutions adaptées personnalisées. Elle répond aux attentes des voyageurs occasionnels, réguliers, professionnels ou de loisirs et reste désormais un outil indispensable face aux imprévus lors des déplacements à l'étranger.

1.2 Définition et évolution

L'assurance voyage est une forme spécifique d'assurance IARD (Incendie, Accidents et Risques Divers), plus précisément une assurance de dommages à caractère temporaire qui couvre une variété de risques liés aux déplacements. Elle prend en charge les frais médicaux, la perte de bagages, l'annulation de vols, ainsi que d'autres pertes financières pouvant survenir à l'occasion d'un voyage, telles que les maladies, blessures, accidents, retards de transport, ou autres événements imprévus susceptibles de compromettre le séjour.

Les garanties offertes incluent notamment :

- la couverture des accidents personnels (décès de l'assuré, invalidité permanente totale) ;
- le remboursement des frais médicaux et de soins dentaires en cas d'accident ou de maladie ;
- l'évacuation médicale d'urgence ;
- l'indemnisation en cas de retard ou de perte de bagages, perte de passeport, retard de vol, détournement d'avion, cambriolage au domicile pendant l'absence, réduction ou annulation de voyage, correspondance ou départ manqué, et réservations non honorées.

Des polices spécifiques peuvent couvrir des risques additionnels selon les besoins particuliers des voyageurs. Dans ces cas, le montant et l'étendue de la couverture dépendent de la police choisie et du plan sélectionné, moyennant souvent une prime additionnelle.

Il est toutefois important de noter que certaines situations ne sont pas couvertes :

- incidents survenus sous l'emprise d'alcool ou de drogues ;
- traitements alternatifs (ayurvéda, homéopathie) et chirurgies esthétiques ;
- maladies préexistantes, dont le VIH, ou voyages effectués contre avis médical ;
- blessures auto-infligées, suicides, et pratiques de sports extrêmes ;
- pertes d'objets laissés sans surveillance ou de bagages expédiés séparément ;
- pertes de passeport ou d'effets personnels non surveillés ;
- voyages dans des zones à risque élevé (guerre, terrorisme).

Avec l'essor du tourisme international, l'assurance voyage est devenue incontournable pour de nombreux voyageurs. Dans certains pays (Arabie Saoudite, Russie, Chine, Cuba,

etc.), elle est même exigée pour l'obtention d'un visa. Le transport aérien, moteur de la connectivité moderne, renforce cette nécessité, tant pour les particuliers que pour les entreprises et gouvernements.

Selon la United States Travel Insurance Association, les dépenses des Américains en assurance voyage ont atteint environ 4 milliards de dollars en 2018, soit une hausse de 41 % par rapport à 2016. Le nombre de voyageurs couverts a progressé de plus de 10,7 % depuis 2021. La pandémie de COVID-19 a par ailleurs mis en lumière le rôle essentiel de l'assurance voyage : les restrictions sanitaires ont bouleversé les plans mondiaux, laissant de nombreux voyageurs sans protection financière. Forbes (2020) rapporte que 62 % des adultes américains ont subi des annulations de vols, beaucoup ayant perdu de l'argent faute d'assurance. Même après la pandémie, les perturbations liées aux conditions météorologiques ou aux mouvements sociaux continuent de souligner la valeur de l'assurance voyage comme élément central de la planification des déplacements.

1.3 Coût et tarification de l'assurance voyage

Le coût d'une assurance voyage représente généralement entre 4 % et 10 % du prix total du voyage. Par exemple, pour un séjour coûtant 10 000 $ l'assurance peut varier entre 400 $ et 1 000 $. Les primes dépendent de plusieurs facteurs :

- le type de couverture souscrite ;
- l'âge du voyageur (un âge avancé accroît le risque de santé) ;
- la destination (en Europe ou hors Europe, selon le niveau de risque du pays) ;
- la durée du voyage (les assureurs considèrent un séjour de plus de trois mois comme une expatriation) ;
- le type de voyage (solo, familial, tour du monde, séjour balnéaire) ;
- les options choisies (assurance bagages, couverture pour sports extrêmes, etc.) ;
- le nombre de personnes assurées (conjoint, enfants à charge, parfois parents et grands-parents).

Des avenants spécifiques sont souvent proposés aux voyageurs d'affaires, aux sportifs ou aux expatriés pour adapter la couverture à des besoins particuliers.

1.4 Fonctionnement de l'assurance voyage

L'assurance voyage peut être souscrite en ligne auprès d'agences de voyages, de prestataires de services (compagnies aériennes, croisiéristes), de compagnies d'assurance privées, ou de courtiers, souvent au moment de réserver un vol, un hébergement ou une location de voiture. Parmi les acteurs majeurs du secteur figurent AIG Travel, Berkshire Hathaway Travel Protection, Generali Global Assistance, GeoBlue, et Nationwide, entre autres.

On distingue généralement deux types de couverture :

- Couverture primaire : l'assurance voyage intervient en premier, remboursant directement l'assuré sans attendre qu'il fasse une réclamation auprès d'une autre

couverture (compagnie aérienne, assurance automobile, etc.), ce qui évite des hausses potentielles de primes.
— Couverture secondaire : l'assuré doit d'abord demander une indemnisation à une autre couverture, et l'assurance voyage intervient uniquement en complément.

1.5 La nécessité d'anticiper les sinistres pour une gestion optimale

La nécessité de prédire les sinistres en assurance voyage réside dans le rôle central que joue cette activité pour garantir l'équilibre économique des compagnies d'assurance. Anticiper les sinistres permet non seulement de mieux calibrer les primes d'assurance, en assurant qu'elles soient à la fois compétitives et suffisantes pour couvrir les risques, mais aussi de renforcer les mécanismes de gestion des réserves et de solvabilité. Dans un secteur où les événements couverts peuvent être fortement imprévisibles — comme les accidents, les urgences médicales à l'étranger ou les annulations de voyage — disposer de modèles prédictifs robustes aide à réduire l'incertitude et à optimiser les ressources. Cela bénéficie non seulement aux assureurs, qui améliorent leur rentabilité, mais aussi aux assurés, qui profitent de produits mieux adaptés à leurs besoins et à leurs profils de risque. Enfin, dans un contexte d'internationalisation croissante des mobilités, la prédiction des sinistres devient un levier essentiel pour innover et proposer des garanties plus flexibles et personnalisées.

1.6 Défis et impacts du déséquilibre des classes dans la prédiction des sinistres

Dans le domaine de l'assurance voyage, on se heurte fréquemment à un problème de déséquilibre des classes : les sinistres avérés (annulations, pertes de bagages, urgences médicales, etc.) sont beaucoup moins nombreux que les voyages sans incident. Un ensemble de données est dit déséquilibré lorsqu'une classe (ici, les « non-sinistres ») domine largement l'autre (les « sinistres ») (Somasundaram & Reddy, 2016). Ce déséquilibre complique considérablement l'apprentissage des modèles prédictifs, car de nombreuses techniques classiques ont tendance à privilégier la classe majoritaire.

Le déséquilibre est généralement mesuré par le rapport entre le nombre total de cas sans sinistre et celui des sinistres. Plusieurs caractéristiques des données déséquilibrées rendent la tâche difficile, notamment : le chevauchement des profils (par exemple, certains voyageurs ressemblant à ceux qui font rarement des sinistres), la rareté des cas problématiques, ou encore la présence de bruit dans les données collectées (More & Rana, 2017).

L'apprentissage supervisé, qui constitue la base des modèles prédictifs en assurance, suppose souvent des données équilibrées pour bien fonctionner (Gong & Kim, 2017). Or, ici, l'événement à prédire — le sinistre — est rare mais coûteux, et une mauvaise prédiction peut entraîner des pertes financières pour l'assureur. Les modèles classiques ont donc tendance à minimiser les erreurs sur la classe majoritaire (voyages sans problème) tout en

ignorant les erreurs sur la classe minoritaire (sinistres), ce qui réduit la pertinence des prédictions.

Le déséquilibre accentue également le coût computationnel, car les algorithmes peuvent générer un grand nombre de motifs associés à la majorité, sans suffisamment détecter ceux, pourtant cruciaux, qui concernent la minorité (Dong & Bailey, 2012). De plus, la classe minoritaire (les sinistres) est souvent celle qui présente le plus d'intérêt stratégique pour l'entreprise, mais elle est difficile à exploiter faute de données suffisantes (Weiss & Tian, 2008).

Enfin, les métriques de performance classiques (comme l'exactitude globale) deviennent trompeuses en présence de déséquilibre. Dans le cadre de ce travail, il est essentiel d'identifier les bonnes approches pour dépasser les biais induits par les déséquilibres et améliorer la performance prédictive des modèles sur les sinistres, en mettant l'accent sur les techniques spécifiquement conçues pour ce type de problème.

Chapitre 2

Revue des principales approches de résolution du problème de déséquilibre des classes

***

De multiples approches ont été développées pour résoudre le problème de déséquilibre des classes, très courant en pratique. Elles peuvent être catégorisées en deux grandes familles, selon la nature de la stratégie de contournement à ce déséquilibre (Branco et al., 2017 ; Haixiang et al., 2017 ; He & Garcia, 2009 ; Kaur et al., 2020). La première famille d'approches est qualifiée de "data-level" ou "d'externe" : elle consiste à modifier l'ensemble de données d'origine par certains mécanismes, afin de réduire le déséquilibre dans la répartition des catégories ou de modifier la distribution des données pour qu'elle soit plus adaptée pour les tâches d'apprentissage ultérieures. La seconde famille est dénommée "algorithm level" ou "interne" : elle repose sur des paramétrages ou modifications des techniques d'apprentissage standards ou sur des développements d'approches spécifiques, afin qu'elles soient en capacité d'identifier plus précisément la classe minoritaire

2.1 Aperçu des principales approches "data-level"

Ces approches consistent en une étape de pré-traitement des données initiales, avant l'entraînement du modèle. Elles se décomposent en deux familles : d'une part les méthodes d'échantillonnage des données, d'autre part les méthodes basées sur la réduction de dimension, généralement par extraction et sélection de caractéristiques ("features" en anglais). Elles présentent l'avantage de pouvoir être utilisées quel que soit le type de classifieur employé ensuite.

Echantillonnage des données

L'idée est de sélectionner une partie des instances de la base d'origine, soit aléatoirement,

soit en utilisant une méthode heuristique particulière, afin de réduire le déséquilibre des classes. On peut procéder à un sous-échantillonnage, en supprimant un certain nombre d'exemples de la classe majoritaire, ou à un sur-échantillonnage, en répliquant ou en ajoutant des points de la catégorie minoritaire.

Pour sur-échantillonnage, elles regroupent les méthodes telles que : Tirage aléatoire uniforme et le Synthetic Minority Over-sampling Technique (SMOTE). On distingue une large extension de la méthode SMOTE (Plus de 85 variantes et extensions de la technique SMOTE) comme le Bordeline-SMOTE, Adaptative Synthetic Sampling (ADASYN), Safe-Level SMOTE, Random Oversampling Examples (ROSE), Majority Weighted Minority Oversampling Technique (MWMOTE), ect.

Dans une étude sur la prédiction améliorée de la réadmission diabétique basée sur une machine à vecteur de soutien, (Cui et al., 2018) la méthode SMOTE a été utilisée pour gérer le déséquilibre des données, suivie de la sélection de caractéristiques hybrides et de l'optimisation SVM à l'aide d'algorithmes génétiques. Les résultats ont montré une précision de 81,02%, une sensibilité de 82,89% et une spécificité de 79,23%, ce qui a surpassé diverses autres méthodes telles que le score LACE, la régression logistique, le bayésien naïf, l'arbre de décision et le réseau neuronal avancé dans l'identification des patients à risque de réadmission.

Une étude similaire sur la validité de l'apprentissage automatique dans la détection de l'appendicite compliquée dans un cadre aux ressources limitées (résultats du Vietnam), (Phan-Mai et al., 2023) a révélé que parmi les différents modèles de machine learning permettant de détecter l'appendicite compliquée, le Gradient Boosting (GB) avait la validité la plus élevée, avec des valeurs AUC et de précision d'environ 0,8 ou plus, avant et après l'application de SMOTE pour équilibrer les données.

Muhammad AbdullahAish (Aish, 2024) dans une Modélisation prédictive des accidents vasculaires cérébraux à l'aide d'une approche ADASYN-RF pour les données déséquilibrées, puis en procédant par des expériences « trois algorithmes différents ont été explorés : RF, KNN et LR, combinés à trois méthodes de suréchantillonnage telles que SMOTE, ADASYN et ROSE », parviennent à des résultats montrant la très bonne performance de l'algorithme de ADASYN-RF sur l'ensemble de données de prédiction de l'AVC cérébral. En outre, l'AUC est une meilleure mesure pour déterminer la meilleure méthode de classification. Le processus discuté aidera à mieux prédire l'AVC cérébral en utilisant ADASYN-RF approche.

De ses méthodes de sous-échantillonage, on retrouve : Tirage aléatoire uniforme, Near-Miss, Condensed Nearest Neighbor (CNN), Random Undersampling, Edited Nearest Neighbor (ENN), Neighborhood Cleaning Rule (NCL), Tomek Links (TL), One-Sided Selection (OSS), K-medoids, Class Purity Maximization (CPM), Under-Sampling Based on Clustering (SBC), ect.

Afin d'effectuer la classification des données de déséquilibre à l'aide de la méthode Condensed Nearest Neighbor (CNN) du plus proche voisin, Siddappa & Kampalappa (2019), grâce aux résultats expérimentaux, ont montré que l'Ada-CNN proposé atteignait une précision de 93,22% et une sensibilité de 89,34% pour les ensembles de données sur le diabète, tandis qu'une précision de 100% et une sensibilité de 55,55% pour l'ensemble de données sur l'échec du POP. L'Ada-CNN, basé sur MLP, a été confronté aux problèmes d'adaptabilité des ensembles de données à grande échelle en raison du choix adaptatif des

valeurs k.

Une combinaison d'un sur-échantillonnage et d'un sous-échantillonnage permet d'obtenir une approche mixte qui peut donner de meilleurs résultats que l'un ou l'autre pris isolément. Quelques représentants de ce type d'approches mixtes sont : SMOTE+TL, SMOTE+TOMEK, SMOTE+ENN, Agglomerative Hierarchical Clustering (AHC) et le Selective Pre-Processing of Imbalance Data (SPIDER).

H. Hairani, A. Anggrawan & D. Priyanto dans leur recherchent (Hairani & Priyanto, 2023) utilisent une approche d'échantillonnage hybride (SMOTE Tomek Link) avec la méthode Random Forest pour prédire le diabète. Dans le même temps, les résultats montrent que la méthode d'échantillonnage hybride (SMOTE-Tomek Link) augmente la précision de la méthode de la forêt aléatoire par rapport à SMOTE et Tomek Link séparément. A. I. ElSeddawy, F. K. Karim, A. M. Hussein et al, dans leur recherche visant à prédir le risque de diabète à l'aide d'une approche d'échantillonnage hybride (SMOTE-Tomek Link) avec la méthode ANN, montrent que l'utilisation de l'échantillonnage hybride SMOTE-Tomek Link est meilleure que SMOTE seul, avec une précision de 92%.

Réduction de la dimension par extraction ou sélection de caractéristiques

Les approches de réduction de dimension consistent souvent en un processus de prétraitement des données qui permet de supprimer les informations redondantes et bruitées. Elles sont généralement classées en deux familles :

- l'extraction de caractéristiques, qui permet de créer de nouveaux prédicteurs, en utilisant une combinaison des caractéristiques de l'espace de départ, ou plus généralement une transformation effectuant une réduction du nombre de dimensions ;
- la sélection de caractéristiques, qui regroupe les algorithmes permettant de déterminer un sous ensemble des d variables d'entrée X pertinent à utiliser pour construire le modèle de classification.

2.2 Apercu des principales approches "algorithm level"

Pour contourner le problème de déséquilibre des classes, les approches "algorithm-level" constituent une alternative aux solutions "data-level". Alors que ces dernières modifient l'ensemble d'apprentissage pour combattre le déséquilibre, les approches "algorithm-level" cherchent à modifier la procédure d'entraînement des classifieurs standards ou à proposer de nouveaux modèles précisément développés pour résoudre le déséquilibre. Ces approches internes ne créant pas de perturbation dans la distribution des données d'origine, elles sont plus facilement adaptables aux différents cas de figure de déséquilibre des classes, au prix par contre d'être spécifiques à un type donné de classifieur. Évidemment, ceci nécessite une compréhension fine du principe de chaque algorithme, de façon à identifier dans sa construction ce qui le pousse à favoriser par défaut la classe majoritaire et c'est une des raisons pour lesquelles ces approches sont moins populaires que les approches "data level".

Une approche courante consiste à ajuster les seuils de décision des classificateurs probabilistes afin d'abaisser la barrière pour prédire une observation comme appartenant à la

classe minoritaire (Dal Pozzolo et al., 2015). Cela permet d'augmenter la sensibilité aux cas rares, bien que parfois au détriment de la précision globale.

Une autre stratégie est le cost-sensitive learning, qui intègre des coûts différenciés dans la fonction de perte des algorithmes pour pénaliser davantage les erreurs sur la classe minoritaire (Elkan, 2001). Cette méthode a été appliquée avec succès à divers modèles, tels que les machines à vecteurs de support pondérées (Lin et al., 2002), les réseaux de neurones à perte pondérée (Zhou & Liu, 2006), et les modèles d'ensemble comme AdaBoost avec ajustement des poids (Sun et al., 2007).

Des algorithmes spécifiques ont également été développés pour traiter le déséquilibre, notamment les Balanced Random Forests (Chen, Liaw & Breiman, 2004), où chaque arbre est construit à partir d'un sous-échantillon équilibré, ainsi que les méthodes d'ensemble telles qu'EasyEnsemble et BalanceCascade, qui combinent plusieurs modèles entraînés sur des sous-jeux rééquilibrés (Liu, Wu & Zhou, 2009).

Enfin, certaines approches visent à optimiser directement des métriques robustes au déséquilibre, comme l'AUC-ROC, le F1-score ou le rappel, au lieu de maximiser uniquement l'accuracy (Saito & Rehmsmeier, 2015). Ces métriques offrent une évaluation plus fiable des performances dans les contextes déséquilibrés, notamment lorsque l'identification des cas rares est prioritaire.

Ces techniques présentent l'avantage de ne pas modifier les données initiales, ce qui limite l'introduction de bruit artificiel. Cependant, elles nécessitent une sélection rigoureuse des hyperparamètres et une bonne connaissance des particularités des algorithmes employés pour obtenir des résultats optimaux (Fernández et al., 2018). Dans ce mémoire, ces différentes stratégies seront explorées afin de relever les défis posés par la prédiction des sinistres rares en assurance voyage.

Chapitre 3

Présentation des données et méthodologie

***

La présente étude s'inscrit dans un contexte où les entreprises d'assurance voyage cherchent à optimiser leurs modèles prédictifs des sinistres en s'appuyant sur les techniques avancées de machine learning (ML). Toutefois, ces modèles se heurtent à une problématique fondamentale : la rareté des sinistres au sein des portefeuilles d'assurance voyage, engendrant un déséquilibre marqué des classes dans les jeux de données. Cette configuration compromet l'efficacité des modèles de classification qui tendent à favoriser la classe majoritaire (absence de sinistre) au détriment de la classe minoritaire (sinistre déclaré), comme l'ont démontré He et Garcia (2009) dans leurs travaux sur les données déséquilibrées.

L'analyse s'appuie sur un corpus de données provenant d'une société spécialisée dans les services d'assurance voyage établie à Singapour. Ces données comprennent 63 326 observations et 11 variables, structurées comme suit :

Caractéristiques de l'agence d'assurance

- Agency : Dénomination de l'agence d'assurance voyage.
- Agency_Type : Typologie de l'agence d'assurance voyage.
- Distribution_Channel : Canal de distribution employé pour la commercialisation des produits d'assurance.

Caractéristiques du contrat d'assurance

- Product_Name : Dénomination du produit d'assurance voyage souscrit.
- Duration : Durée du séjour assuré.
- Net_Sales : Montant des ventes de la police d'assurance.
- Commission_in_value : Commission perçue par l'agence sur la commercialisation de la police d'assurance.

Caractéristiques de l'assuré

- Gender : Genre de l'assuré.
- Age : Âge de l'assuré.
- Destination : Destination du voyage.

Variable cible (à prédire)

- Claim : Statut de la réclamation (présence ou absence de sinistre déclaré).

3.1 Statistiques descriptives

Avant de procéder aux analyses avancées, il est essentiel de réaliser une étude statistique descriptive afin de dresser un portrait général du jeu de données. Cette étape permet d'identifier les principales caractéristiques des variables, de repérer d'éventuelles anomalies (valeurs manquantes, valeurs extrêmes, distributions asymétriques) et de mieux comprendre la structure sous-jacente des données.

3.1.1 Analyse des variables catégorielles

Déséquilibre sévère de la variable Claim

La variable Claim est fortement déséquilibrée : 98,32% des contrats n'ont pas donné lieu à une réclamation contre 1,68% de sinistres déclarés. Ce déséquilibre, courant en assurance, pose un défi pour l'apprentissage automatique, car un modèle non ajusté risque de prédire majoritairement l'absence de sinistre, rendant la classification des sinistres inefficace.

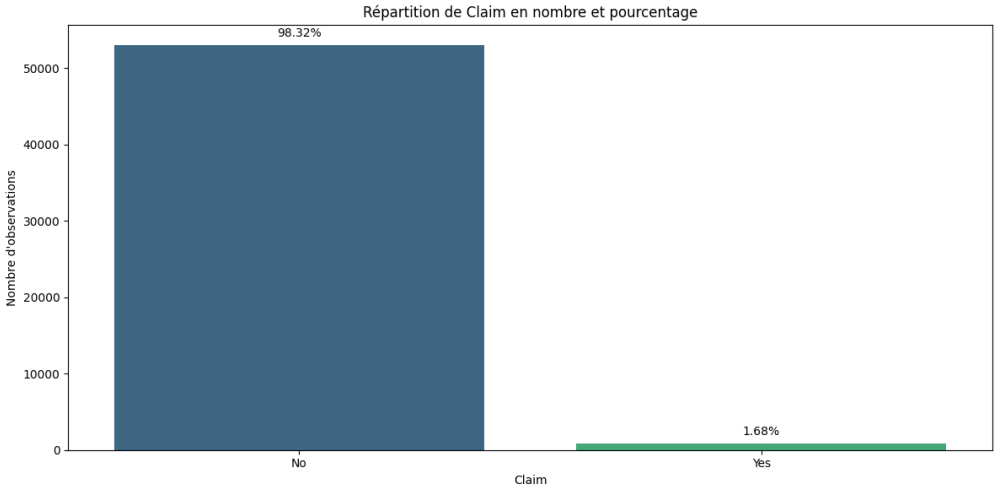

FIGURE 3.1 – Sinistralité

Concentration oligopolistique des agences

L'examen de la distribution des agences révèle une forte concentration des contrats auprès d'un nombre restreint d'acteurs : EPX domine largement avec 53,26% des observations, suivie de C2B (14,34%) et CWT (14,30%), qui affichent des parts de marché presque équivalentes mais nettement inférieures. JZI occupe une position significative avec 11,02%, tandis que SSI, bien plus marginale (1,74%), reste néanmoins parmi les principaux contributeurs. Ensemble, ces cinq agences concentrent plus de 94% du jeu de données, soulignant une structure de marché oligopolistique susceptible d'influencer l'interprétation des analyses prédictives selon l'origine institutionnelle des contrats.

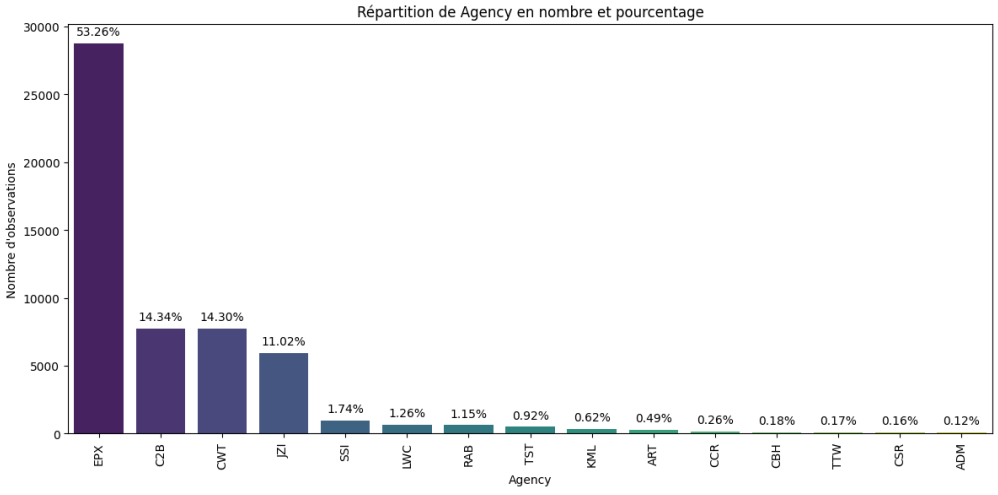

FIGURE 3.2 – Répartition des agences

Prédominance des agences de voyage

La répartition selon le type d'agence montre une prédominance des agences de voyage (71,26%) par rapport aux compagnies aériennes (28,74%). Cette répartition indique que la majorité des contrats d'assurance sont souscrits via des agences de voyage.

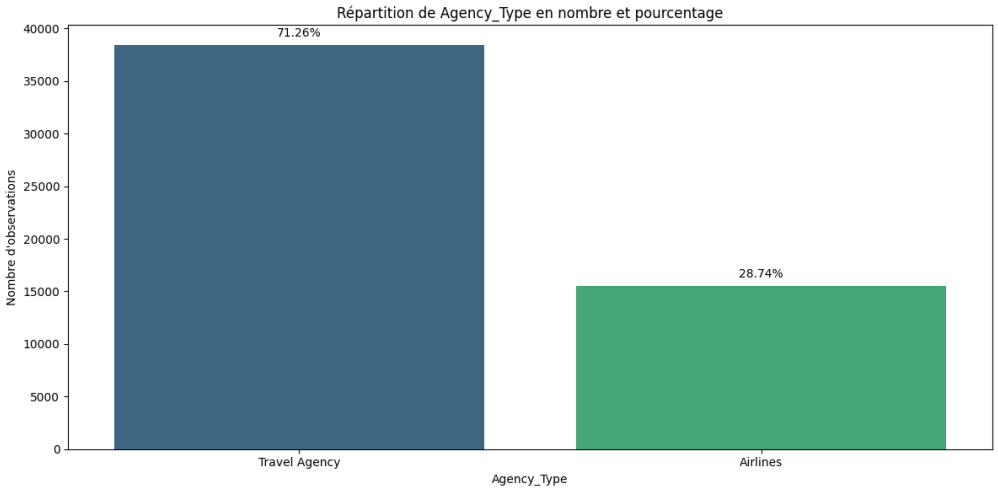

FIGURE 3.3 – Répartition des type d'agence

Domination écrasante du canal en ligne

La distribution des contrats est quasiment exclusivement en ligne (98,22%), avec une part infime pour le canal offline (1,78%). Cette dominance du digital suggère que l'essentiel des souscriptions se fait via des plateformes en ligne, ce qui peut influencer les comportements des assurés et la gestion des sinistres.

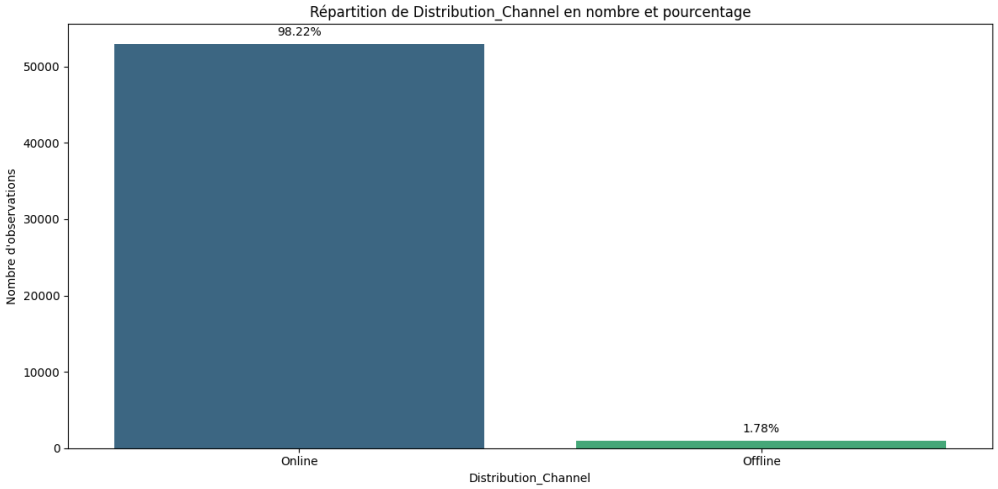

FIGURE 3.4 – Chaîne de distribution

Forte concentration des contrats

L'analyse des cinq produits les plus représentés met en évidence une forte concentration des observations. Le *Cancellation Plan* domine nettement avec 35,19% des occurrences, suivi par le *2 Way Comprehensive Plan* (25,60%) et le *Rental Vehicle Excess Insurance* (18,58%). Les deux derniers du classement, *Basic Plan* (11,92%) et *Bronze Plan* (8,71%), complètent ce top 5. Ensemble, ces produits totalisent une part substantielle des contrats enregistrés, reflétant une préférence marquée des clients pour certaines offres. Cette hiérarchisation

des produits doit être prise en compte dans l'analyse prédictive, car elle peut influencer la structure des données et la performance des modèles.

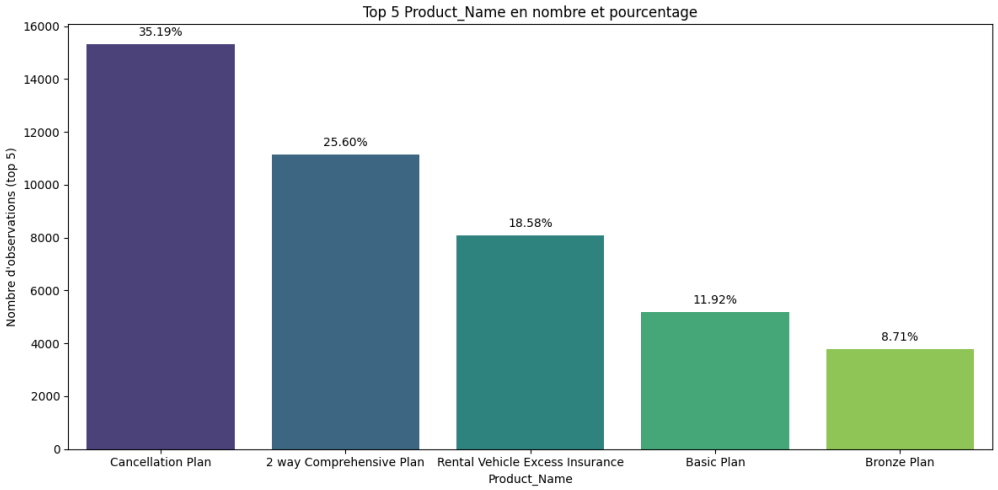

FIGURE 3.5 – Les contrats les plus représentés

Genre : 69,32% de valeurs manquantes, traitées comme modalité distincte selon les recommandations de Sperrin et al. (2020) et van Buuren (2018)

La répartition des assurés selon le genre montre que près de 69,32% des observations n'ont pas d'information renseignée sur le genre (catégorie «Missing»), ce qui représente une part écrasante de l'échantillon. Les individus identifiés comme M (hommes) et F (femmes) sont en proportions presque équivalentes, avec respectivement 15,48% et 15,20% des observations, soit environ un huitième chacun. Dans cette analyse, les valeurs manquantes de la variable Gender ont été considérées comme une troisième modalité distincte. Le traitement des valeurs manquantes comme une modalité distincte est une approche couramment utilisée lorsque l'absence d'information peut elle-même contenir un signal pertinent (Sperrin et al., 2020). En effet, selon van Buuren (2018), considérer les valeurs manquantes comme une catégorie séparée permet non seulement de préserver l'intégralité des données, mais aussi d'éviter les biais qui pourraient être introduits par une imputation inadéquate, notamment lorsque les mécanismes de données manquantes ne sont pas aléatoires.

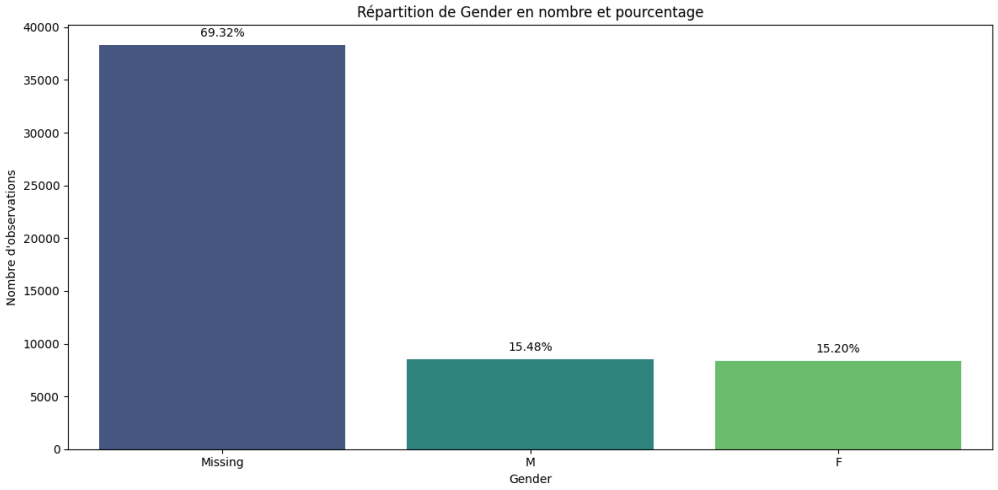

FIGURE 3.6 – Sexe du souscripteur

Concentration majeure sur quelques pays, avec longue traîne de destinations rares, un enjeu clé pour la modélisation

L'analyse de la variable Destination met en évidence une répartition très déséquilibrée entre les pays. Un petit nombre de destinations, comme Singapour, Thaïlande, Chine, États-Unis, Australie, Malaisie, Indonésie, Hong Kong et Vietnam, concentre la majorité des observations, ce qui traduit leur poids dominant dans le jeu de données. D'autres pays, tels que Philippines, Japon, Royaume-Uni, Inde, Canada et Émirats arabes unis, apparaissent également de façon notable, mais à un niveau moins élevé. Enfin, une longue traîne de destinations rares est présente, composée de nombreux pays dont la fréquence est marginale. Cette hétérogénéité est importante à considérer car elle pourrait influencer les analyses ultérieures, notamment dans les phases de modélisation ou de segmentation, où les classes rares risquent d'être sous-représentées et donc mal apprises par les modèles statistiques ou d'apprentissage automatique.

FIGURE 3.7 – Pays de destination de l'assuré

3.1.2 Analyse des variables quantitatives

L'analyse de la variable Duration révèle une distribution fortement asymétrique à droite, avec une majorité de valeurs faibles et quelques valeurs extrêmes dépassant 5000 jours. Le boxplot met en évidence la présence d'outliers significatifs, tandis que le Q-Q plot confirme un écart important par rapport à la normalité.

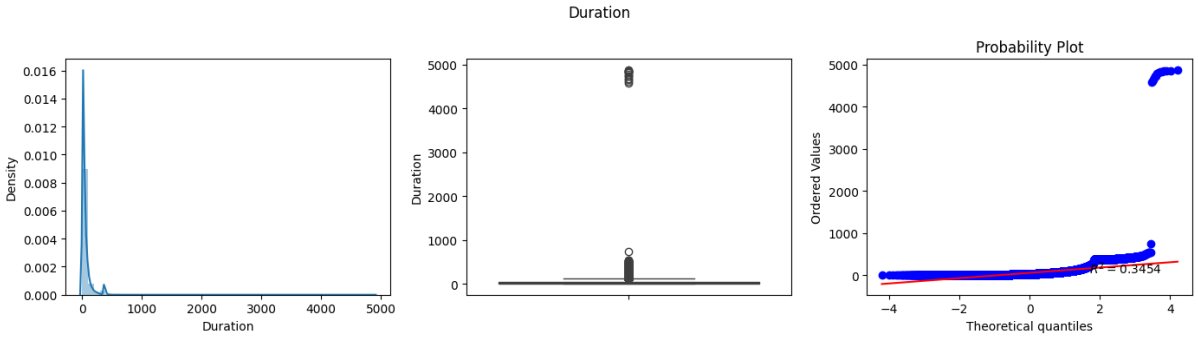

FIGURE 3.8 – Distribution de la durée du voyage

Les ventes nettes présentent une distribution asymétrique à droite, avec une concentration des valeurs autour de 0 à 100 et quelques valeurs extrêmes dépassant 600. Le boxplot confirme la présence de nombreux outliers, et le Q-Q plot montre un fort écart par rapport à la normalité.

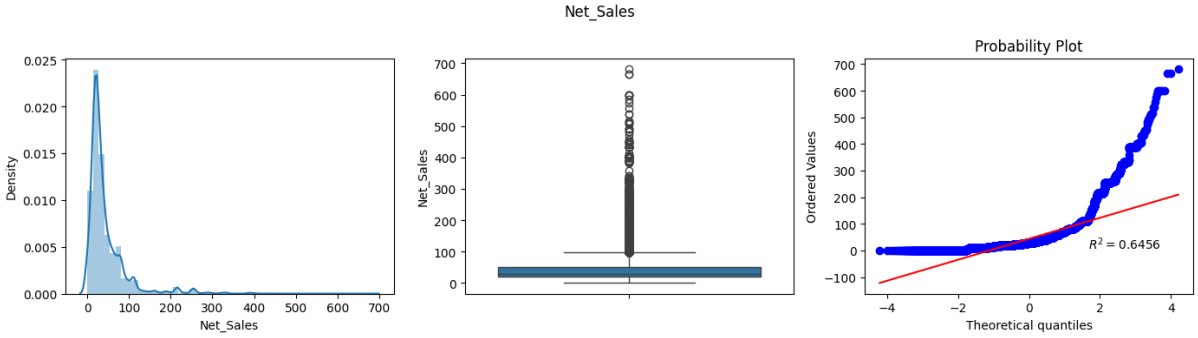

FIGURE 3.9 – Distribution des ventes nettes

La variable désignant la commission présente une distribution fortement asymétrique à droite, avec une concentration majoritaire des valeurs près de zéro. Le graphique de densité montre un pic prononcé dans les faibles valeurs et une longue queue vers la droite. La boîte à moustaches révèle de nombreuses valeurs aberrantes au-dessus de 250. Le QQ-plot confirme cette non-normalité avec un R^2 de 0,5551 et une courbe en "J" caractéristique. Cette structure indique que la plupart des contrats génèrent de faibles commissions, tandis qu'un petit nombre produit des commissions élevées.

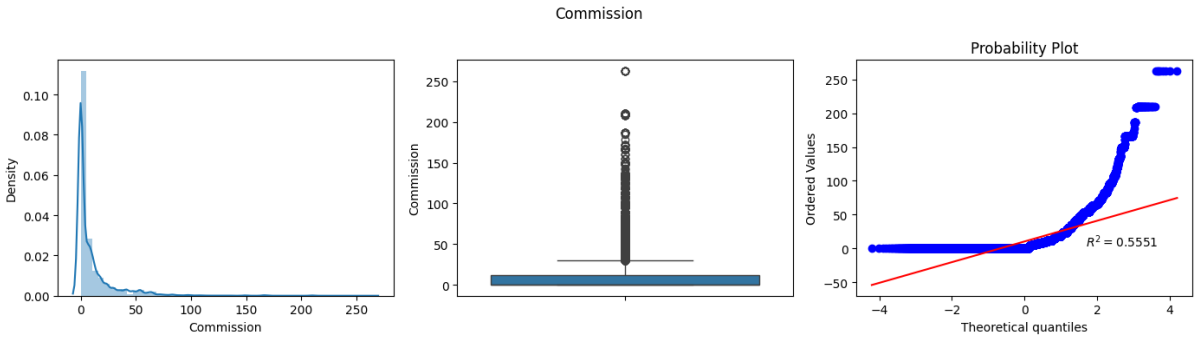

FIGURE 3.10 – Distribution des commissions

La variable "Age" montre une distribution atypique avec un pic très prononcé vers 35-36 ans. Le graphique de densité révèle également des pics secondaires vers 45-50 ans. La boîte

à moustaches indique une médiane autour de 40 ans, avec un écart interquartile d'environ 35 à 45 ans et quelques valeurs aberrantes aux extrémités. Le QQ-plot présente un R^2 de 0,8993, suggérant une approximation acceptable de la normalité malgré la concentration marquée à 35-36 ans. Cette distribution pourrait indiquer une surreprésentation des voyageurs d'affaires d'âge moyen dans le portefeuille d'assurance étudié.Cette distribution, avec son pic prononcé à 35-36 ans, pourrait refléter une forte proportion de voyageurs d'affaires ou une caractéristique spécifique des produits d'assurance commercialisés qui attirent particulièrement cette tranche d'âge.

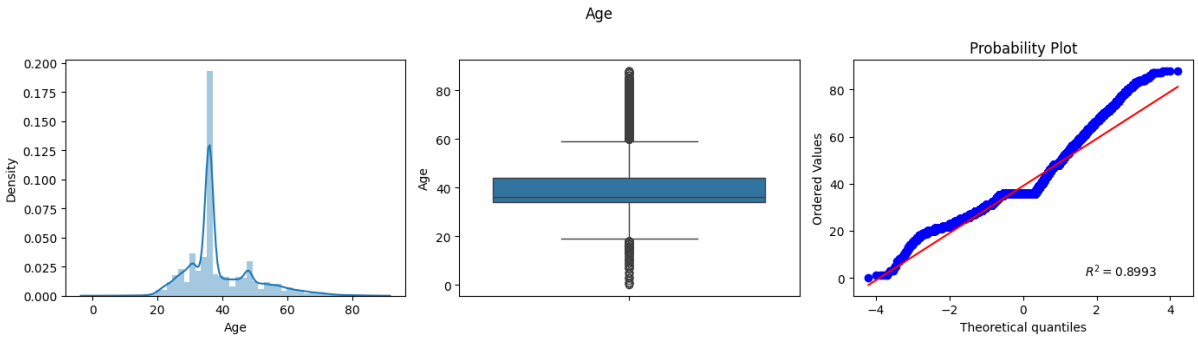

FIGURE 3.11 – Distribution de l'âge des souscripteurs

3.2 Preprocessing

Le pré-processing permet notamment de nettoyer les données, de traiter les valeurs manquantes, de normaliser ou standardiser les variables numériques, et de transformer les variables catégorielles en représentations numériques exploitables (Hastie, Tibshirani & Friedman, 2009). Sans cette préparation, les modèles risquent non seulement d'être moins performants, mais aussi de produire des résultats biaisés ou peu interprétables (Kuhn & Johnson, 2013).

Par conséquent, cette étape préalable n'est pas une simple formalité, mais une condition indispensable pour garantir la robustesse, la fiabilité et la pertinence des analyses prédictives qui suivent.

3.2.1 Transformation de Yeo-Johnson

Les variables telles que la durée du voyage, les ventes nettes et l'âge du client présentent une asymétrie marquée et la présence de valeurs extrêmes, comme l'indiquent leurs distributions et leurs diagrammes de dispersion. Une telle structure peut nuire aux performances des modèles statistiques et aux tests basés sur l'hypothèse de normalité (Yeo, I. K., & Johnson, R. A., 2000). Pour remédier à cela, la transformation de Yeo-Johnson est appliquée.

Formulation Théorique

La transformation de Yeo-Johnson est une extension de la transformation de Box-Cox, permettant d'améliorer la normalité et la symétrie d'une variable, y compris lorsque celle-ci contient des valeurs négatives ou nulles. Elle est particulièrement utile pour traiter

les distributions asymétriques et réduire l'effet des valeurs extrêmes, ce qui améliore les performances des modèles de machine learning qui supposent souvent une distribution gaussienne des variables.

Soit y la variable à transformer et λ un paramètre de transformation. La transformation de Yeo-Johnson est définie comme suit :

$$T(y,\lambda) = \begin{cases} \frac{(y+1)^{\lambda}-1}{\lambda}, & \text{si } y \geq 0 \text{ et } \lambda \neq 0 \\ \log(y+1), & \text{si } y \geq 0 \text{ et } \lambda = 0 \\ \frac{-(|y|+1)^{(2-\lambda)}-1}{2-\lambda}, & \text{si } y < 0 \text{ et } \lambda \neq 2 \\ -\log(|y|+1), & \text{si } y < 0 \text{ et } \lambda = 2 \end{cases}$$

Le paramètre λ est estimé de manière à optimiser la normalité des données. Contrairement à Box-Cox, Yeo-Johnson est applicable aux valeurs nulles et négatives, ce qui le rend plus flexible dans des contextes variés, notamment en analyse actuarielle et en économétrie des sinistres.

La log-vraisemblance associée à la transformation de Yeo-Johnson permet d'estimer le paramètre λ en maximisant la normalité des données transformées. Elle est définie comme suit :

$$\log L(\lambda) = -\frac{n}{2}\log\left(\frac{1}{n}\sum_{i=1}^{n}(T(y_i,\lambda)-\bar{T})^2\right) + (\lambda-1)\sum_{y_i\geq 0}\log(y_i+1) + (1-\lambda)\sum_{y_i<0}\log(|y_i|+1)$$

où : n est le nombre d'observations, $T(y_i,\lambda)$ est la transformation de Yeo-Johnson appliquée à y_i, $\bar{T}$ est la moyenne des valeurs transformées.

L'estimation de λ repose sur la maximisation de cette fonction de log-vraisemblance. Une fois λ déterminé, la transformation est appliquée aux données pour les rapprocher d'une distribution normale, améliorant ainsi leur interprétabilité et leur compatibilité avec les modèles statistiques.

Application sur les variables quantitatives

La transformation de Yeo-Johnson appliquée à la durée du voyage a permis une amélioration significative de sa distribution. Initialement fortement asymétrique, la densité montre désormais une forme plus symétrique et proche d'une distribution normale. Le boxplot révèle une meilleure répartition des valeurs avec une réduction des valeurs extrêmes, bien que quelques outliers subsistent. L'analyse du Q-Q plot confirme cette normalisation, avec un alignement quasi parfait des points sur la droite théorique et un coefficient de détermination **$R^2 = 0.9948$**, indiquant une très bonne adéquation à la loi normale.

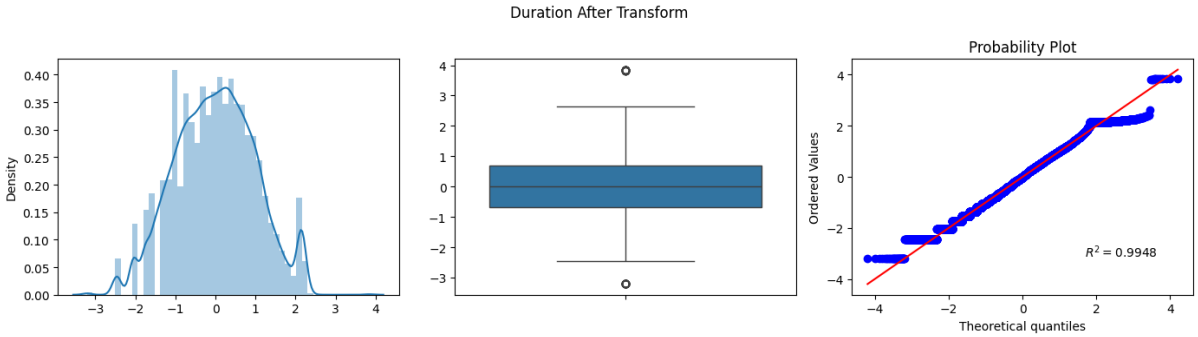

FIGURE 3.12 – Distribution de la durée du voyage après transformation

Après application de la transformation de Yeo-Johnson sur les ventes nettes, on observe une amélioration de la distribution, bien que certaines irrégularités persistent. La densité présente une forme plus symétrique, mais conserve quelques pics. Le boxplot indique une meilleure répartition des valeurs, bien que plusieurs valeurs extrêmes soient toujours présentes. Le Q-Q plot montre un alignement relativement bon des points avec la droite théorique, suggérant une normalisation partielle de la distribution. Le coefficient $R^2 = 0.9523$ confirme cette amélioration, bien qu'elle soit moins marquée que pour la variable Duration.

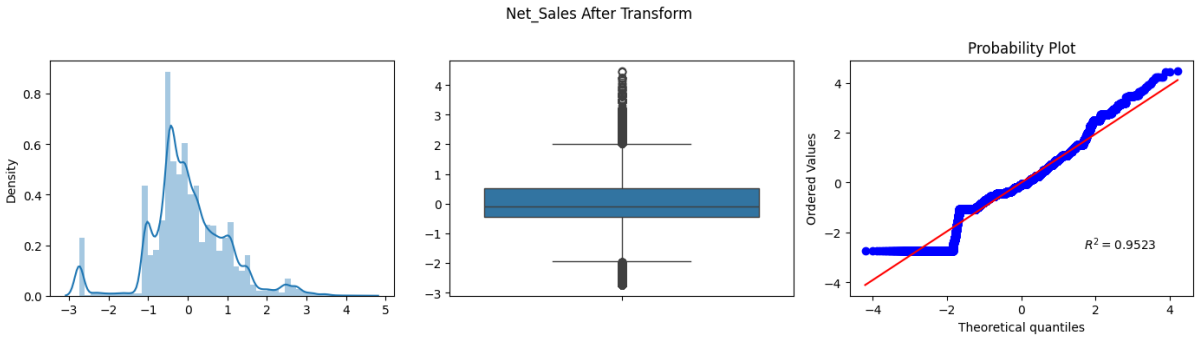

FIGURE 3.13 – Distribution des ventes nettes après transformation

L'analyse des distributions après application de la transformation de Yeo-Johnson sur la variable Commission met en évidence plusieurs améliorations notables. La densité observée montre une distribution désormais plus symétrique, bien que deux zones distinctes subsistent, suggérant la présence possible de sous-populations au sein des données. Le diagramme en boîte révèle une réduction significative des valeurs extrêmes, traduisant un resserrement de l'amplitude et une meilleure concentration des observations autour de la médiane. Enfin, le graphique Q-Q indique un alignement nettement amélioré des valeurs transformées avec la droite théorique de normalité, même si des écarts persistent aux extrémités. Avec un coefficient de détermination $R^2 = 0{,}7515$, la transformation a permis d'augmenter substantiellement la normalité des données sans toutefois atteindre une conformité parfaite. Ces résultats justifient pleinement l'utilisation de la transformation de Yeo-Johnson pour renforcer la robustesse des analyses statistiques et économétriques fondées sur cette variable.

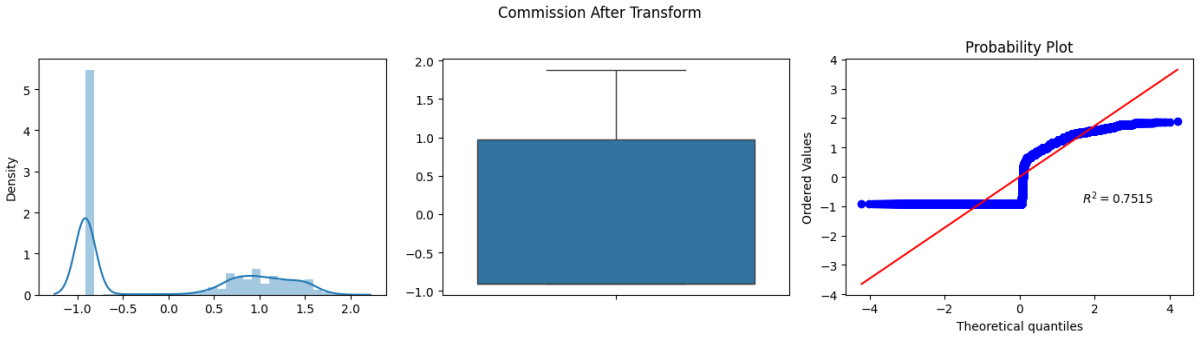

FIGURE 3.14 – Distribution des commissions après transformation

Concernant la variable Age, on constate une réduction de l'asymétrie par rapport à la distribution initiale, bien que le pic principal demeure très marqué. Le boxplot révèle encore plusieurs valeurs extrêmes, en particulier du côté négatif. Le Q-Q plot présente un alignement partiel des points sur la droite de normalité, avec un $R^2 = 0.9393$ qui indique une amélioration notable.

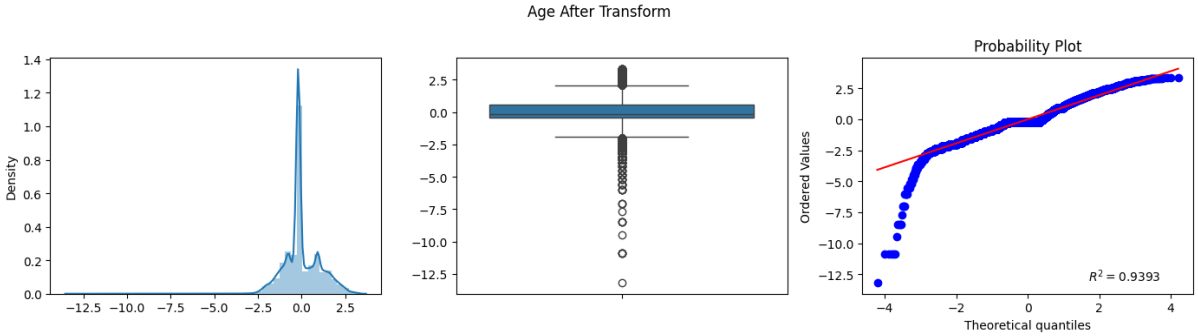

FIGURE 3.15 – Distribution de l'âge des souscripteurs après transformation

3.2.2 Traitement des valeurs aberrantes

Afin de détecter et traiter les valeurs aberrantes, une technique standard basée sur la méthode des écarts interquartiles (Interquartile Range ou IQR) est utilisée. Cette approche est une méthode robuste non paramétrique qui ne fait pas d'hypothèse sur la distribution sous-jacente des données.

Fondement théorique

La méthode IQR pour le traitement des valeurs aberrantes se fonde sur les principes suivants :

1. Calcul des quartiles : Le premier quartile Q1 (25ème percentile) représente la valeur en dessous de laquelle se trouvent 25% des observations. Le troisième quartile Q3 (75ème percentile) représente la valeur en dessous de laquelle se trouvent 75% des observations.

2. Calcul de l'écart interquartile (IQR) : L'IQR est défini comme la différence entre le troisième et le premier quartile :

$$\text{IQR} = Q3 - Q1$$

3. Définition des limites (whiskers) pour les valeurs aberrantes : La limite inférieure (lower whisker) est définie par :

$$\text{Limite inférieure} = Q1 - 1.5 \times \text{IQR}$$

La limite supérieure (upper whisker) est définie par :

$$\text{Limite supérieure} = Q3 + 1.5 \times \text{IQR}$$

4. Identification des valeurs aberrantes : Toute valeur inférieure à la limite inférieure ou supérieure à la limite supérieure est considérée comme une valeur aberrante.

Traitement

Une technique appelée "winsorisation" est utilisée. Pour chaque valeur x dans la colonne de données, la transformation appliquée est :

$$x' = \begin{cases} \text{Limite inférieure} & \text{si } x < \text{Limite inférieure} \\ x & \text{si Limite inférieure} \leq x \leq \text{Limite supérieure} \\ \text{Limite supérieure} & \text{si } x > \text{Limite supérieure} \end{cases}$$

Cette méthode préserve la distribution générale des données tout en atténuant l'influence des valeurs extrêmes qui pourraient biaiser l'analyse statistique ou les modèles de machine learning.

Justification statistique

La règle du 1.5 × IQR pour définir les valeurs aberrantes est attribuée à John Tukey, pionnier de l'analyse exploratoire des données. Cette règle repose sur des propriétés de la distribution normale :

• Pour une distribution normale, environ 99.3% des données se situent dans l'intervalle [Q1 - 1.5 × IQR, Q3 + 1.5 × IQR].

• Par conséquent, seulement environ 0.7% des observations d'une distribution normale seraient considérées comme des valeurs aberrantes selon cette définition.

3.2.3 Encodage des variables catégorielles

Les variables catégorielles, qui n'ont pas d'ordre naturel ni de relation numérique explicite (Kuhn & Johnson, 2013), ne peuvent pas être directement utilisées par les modèles d'apprentissage automatique classiques, tels que les régressions linéaires, logistiques ou les arbres de décision (James et al., 2013), car ces algorithmes ne traitent pas les chaînes de caractères. Il est donc nécessaire de transformer ces variables en représentations numériques appropriées, par des techniques comme l'encodage one-hot, l'encodage ordinal ou l'utilisation d'embeddings (Hastie, Tibshirani & Friedman, 2009), afin de les rendre compatibles avec les modèles.

Le one-hot encoding est une méthode couramment utilisée pour encoder des variables catégorielles. Soit une variable catégorielle X qui peut prendre k catégories distinctes, c'est-à-dire $X \in \{C_1, C_2, \ldots, C_k\}$, où chaque C_i représente une catégorie spécifique.

Pour encoder cette variable X à l'aide du one-hot encoding, on transforme chaque observation de X en un vecteur binaire de dimension k. Chaque vecteur contient un seul élément égal à 1, correspondant à la catégorie de l'observation, et tous les autres éléments sont égaux à 0. Formellement, si l'observation appartient à la catégorie C_i, alors le vecteur $\mathbf{v}_i \in \mathbb{R}^k$ est défini comme suit :

$$\mathbf{v}_i = [0, 0, \ldots, 1, \ldots, 0]$$

où la position du 1 est l'indice de la catégorie C_i.

De manière plus générale, si $\mathbf{v}_X$ représente le vecteur one-hot encodé associé à une observation $X \in \{C_1, C_2, \ldots, C_k\}$, alors ce vecteur $\mathbf{v}_X \in \mathbb{R}^k$ est défini par :

$$\mathbf{v}_X = [\mathbb{I}(X = C_1), \mathbb{I}(X = C_2), \ldots, \mathbb{I}(X = C_k)]$$

où $\mathbb{I}(X = C_i)$ est la fonction indicatrice qui vaut 1 si $X = C_i$ et 0 sinon.

3.3 Méthodologie

Après avoir effectué les étapes de prétraitement des données (nettoyage, normalisation et encodage), l'analyse a été structurée en plusieurs phases méthodologiques complémentaires, visant à optimiser les performances des modèles d'apprentissage.

Dans un premier temps, une sélection des variables explicatives a été réalisée à l'aide de méthodes dites « intégrées » (embedded methods), exploitant la capacité de certains algorithmes à identifier automatiquement les variables les plus informatives lors de l'entraînement. Cette étape a permis de réduire la dimensionnalité du jeu de données tout en conservant les variables les plus pertinentes.

Une fois la sélection effectuée, le jeu de données a été scindé en deux sous-ensembles distincts : un ensemble d'entraînement (train set) utilisé pour l'apprentissage des modèles, et un ensemble de test (test set) réservé à l'évaluation finale des performances sur des données jamais vues.

Ensuite, pour traiter le déséquilibre entre les classes, différentes techniques de rééchantillonnage au niveau des données (data-level methods) ont été appliquées. Ces techniques incluent le suréchantillonnage (telles que SMOTE, ADASYN, Random Over Sampling), le sous-échantillonnage (Random Under Sampling, Condensed Nearest Neighbor), ainsi que des méthodes combinées (comme SMOTEENN et SMOTE+Tomek), dans le but d'améliorer la représentativité de la classe minoritaire.

Enfin, plusieurs modèles d'apprentissage automatique ont été testés, combinant des méthodes de base (algorithm-level) comme la régression logistique, les arbres de décision ainsi que des méthodes ensemblistes plus avancées telles que la forêt aléatoire (Random Forest), le gradient boosting (XGBoost) et les réseaux de neurones artificiels. Cette diversité algorithmique a permis d'évaluer les performances sous différents angles et de sélectionner le modèle le plus performant selon les critères choisis.

3.3.1 Sélection des variables

La sélection de variables est une étape cruciale en apprentissage automatique. Elle vise à identifier les variables explicatives les plus pertinentes pour améliorer la performance des modèles, réduire la complexité computationnelle et éviter le sur-apprentissage (overfitting). Plusieurs approches complémentaires ont été combinées pour sélectionner les meilleures variables.

Méthodes utilisées

1. **Corrélation de Pearson**

   La corrélation de Pearson mesure l'intensité de la relation linéaire entre une variable explicative X_i et la variable cible Y. Pour chaque variable, le coefficient est calculé par :

   $$\rho(X_i, Y) = \frac{\text{Cov}(X_i, Y)}{\sigma_{X_i}\sigma_Y}$$

   où σ représente l'écart-type. Les variables avec les coefficients absolus les plus élevés sont sélectionnées. Cette méthode est simple mais ne capture que des relations linéaires.

2. **Régression Logistique pénalisée (L2)**

   Une régression logistique avec régularisation de type Ridge (ℓ_2) est ajustée. Les variables associées aux plus grands coefficients en valeur absolue (après pénalisation) sont sélectionnées via l'approche `SelectFromModel`.

   L'objectif est de minimiser la fonction suivante :

   $$\mathcal{L}(\beta) = -\sum_{i=1}^{n} [y_i \log(\hat{y}_i) + (1 - y_i) \log(1 - \hat{y}_i)] + \lambda \sum_{j=1}^{p} \beta_j^2$$

3. **Forêt Aléatoire (Random Forest)**

   Un classifieur `RandomForest` est entraîné, et l'importance des variables est évaluée par leur contribution à la réduction de l'impureté (par exemple l'indice Gini). Les variables les plus importantes sont ensuite sélectionnées.

   L'importance d'une variable est généralement calculée comme la moyenne de la réduction d'impureté qu'elle produit dans tous les arbres du modèle.

4. **Gradient Boosting via LightGBM**

   Un modèle LightGBM est construit avec des paramètres spécifiés (par exemple, nombre d'estimateurs, taux d'apprentissage, etc.). L'importance des variables est déterminée par :

   — Le nombre de fois qu'une variable est utilisée pour diviser un nœud (*split importance*),
   — ou la contribution au gain d'information (*gain importance*).

   Les variables les plus importantes selon LightGBM sont retenues via `SelectFromModel`.

Procédure de combinaison

Après avoir effectué les sélections individuelles, les résultats sont synthétisés dans un tableau. Pour chaque variable, le nombre de méthodes qui l'ont sélectionnée est comptabilisé.

$$\text{Total sélection} = \sum_{\text{méthodes}} \text{Support de sélection}$$

Les variables ayant été sélectionnées par au moins deux méthodes (Total ≥ 2) sont conservées pour l'analyse finale.

3.4 Train-Test split

Afin d'évaluer la performance des modèles de manière rigoureuse, le dataset a été divisé en deux sous-ensembles : 70% des observations ont été utilisées pour l'apprentissage, tandis que les 30% restants ont servi à tester la capacité de généralisation du modèle sur des données inédites. Ce choix de répartition, couramment adopté dans la littérature (Hastie, Tibshirani, & Friedman, 2009), permet de garantir un bon compromis entre la qualité de l'apprentissage, qui nécessite une base suffisamment riche, et la fiabilité de l'évaluation, qui exige un échantillon test représentatif.

3.5 Rééquilibrage des données

3.5.1 Over sampling avec ROS

Le Sur-échantillonnage aléatoire (Random Over-Sampling, ou ROS) est l'une des méthodes les plus simples et les plus utilisées pour traiter le déséquilibre des classes dans les jeux de données. Elle consiste à augmenter artificiellement la taille de la classe minoritaire en dupliquant aléatoirement certaines de ses observations, jusqu'à atteindre une distribution des classes plus équilibrée.

Algorithme

Soit $X_m \subset \mathbb{R}^p$ l'ensemble des observations appartenant à la classe minoritaire, avec n_m observations, et $X_M \subset \mathbb{R}^p$ l'ensemble de la classe majoritaire, avec n_M observations.

L'objectif est de générer $G = n_M - n_m$ nouvelles observations synthétiques, afin d'égaliser la taille des deux classes.

1. Sélectionner aléatoirement G observations à partir de X_m, avec remise. On note $\tilde{X}_m = \{x_1^*, \ldots, x_G^*\}$, où chaque $x_i^* \in X_m$.
2. Ajouter $\tilde{X}_m$ à l'ensemble initial de la classe minoritaire :

$$X_m^{\text{new}} = X_m \cup \tilde{X}_m$$

3. L'ensemble de données final devient :

$$X^{\text{final}} = X_M \cup X_m^{\text{new}}, \quad \text{avec } |X_M| = |X_m^{\text{new}}|$$

Le ROS permet ainsi de corriger le déséquilibre de classes, tout en conservant la distribution initiale des données minoritaires. Cependant, comme les nouvelles observations ne sont que des copies, cette méthode peut conduire à un sur-apprentissage si le modèle est trop sensible à la redondance des données.

3.5.2 Over sampling avec SMOTE

Le Sur-échantillonnage (Oversampling) avec SMOTE (Synthetic Minority Over-sampling Technique) est une méthode couramment utilisée pour traiter les problèmes de déséquilibre des classes en apprentissage automatique. Contrairement au sur-échantillonnage classique qui duplique simplement les observations minoritaires, SMOTE génère de nouveaux échantillons synthétiques en interpolant entre les observations existantes. Cela permet d'améliorer la généralisation du modèle en réduisant le sur-apprentissage (overfitting).

Algorithme

Soit X_m l'ensemble des observations appartenant à la classe minoritaire, avec n_m observations. Pour chaque point $x_i \in X_m$, SMOTE génère une nouvelle observation synthétique x_{new} en suivant les étapes suivantes :

1. Sélectionner k voisins les plus proches de x_i parmi les autres observations minoritaires, en utilisant une métrique de distance telle que la distance Euclidienne :

$$d(x_i, x_j) = \sqrt{\sum_{l=1}^{p} (x_{i,l} - x_{j,l})^2}, \quad \forall x_j \in X_m, \quad j \neq i$$

2. Choisir aléatoirement un voisin x_{nn} parmi ces k voisins.
3. Générer un nouvel échantillon x_{new} par interpolation linéaire :

$$x_{\text{new}} = x_i + \lambda \cdot (x_{nn} - x_i), \quad \lambda \sim U(0,1)$$

où λ est un facteur aléatoire tiré d'une distribution uniforme $U(0,1)$.

Cette méthode permet de positionner les nouvelles observations aléatoirement le long du segment reliant x_i et x_{nn}, réduisant ainsi le sur-apprentissage (overfitting) et améliorant la distribution de la classe minoritaire.

3.5.3 Over sampling avec ADASYN

Le sur-échantillonnage (Oversampling) avec **ADASYN** (Adaptive Synthetic Sampling) est une méthode avancée pour traiter les problèmes de déséquilibre des classes en apprentissage automatique. Contrairement à SMOTE, qui génère un nombre fixe d'échantillons synthétiques pour chaque observation minoritaire, ADASYN adapte la génération d'échantillons

en fonction de la difficulté de classification des observations minoritaires. Ainsi, plus une observation est entourée d'instances majoritaires, plus elle génère d'échantillons synthétiques. Cette approche vise à concentrer l'effort d'apprentissage sur les zones complexes, améliorant ainsi la performance globale du modèle.

Algorithme

Soit X_m l'ensemble des observations appartenant à la classe minoritaire, avec n_m observations, et X_M celui de la classe majoritaire. ADASYN procède selon les étapes suivantes :

1. Pour chaque observation minoritaire $x_i \in X_m$, déterminer ses k plus proches voisins dans l'ensemble d'apprentissage (comprenant les classes majoritaire et minoritaire).
2. Calculer le nombre de voisins majoritaires k_i parmi ces k voisins.
3. Déterminer le degré de difficulté de classification r_i pour chaque x_i :

$$r_i = \frac{k_i}{k}, \quad 0 \leq r_i \leq 1$$

   Un r_i proche de 1 indique que x_i est entouré principalement d'observations majoritaires, et est donc plus difficile à classifier correctement.
4. Normaliser les coefficients r_i pour obtenir une distribution de densité G_i :

$$G_i = \frac{r_i}{\sum_{j=1}^{n_m} r_j}, \quad \text{tel que } \sum_{i=1}^{n_m} G_i = 1$$

5. Déterminer le nombre total d'échantillons synthétiques à générer :

$$G = (n_M - n_m) \cdot \beta$$

   où $\beta \in [0, 1]$ est un paramètre définissant le niveau de sur-échantillonnage souhaité.
6. Pour chaque observation x_i, générer $g_i = G_i \cdot G$ échantillons synthétiques. Chaque nouvel échantillon est obtenu par interpolation linéaire avec un voisin minoritaire $x_{nn} \in X_m$:

$$x_{\text{new}} = x_i + \lambda \cdot (x_{nn} - x_i), \quad \lambda \sim U(0, 1)$$

3.5.4 Sous-échantillonnage aléatoire (Random Under-Sampling)

Le Sous-échantillonnage aléatoire (Random Under-Sampling, ou RUS) est une technique simple et rapide pour traiter les problèmes de déséquilibre de classes. Elle consiste à réduire la taille de la classe majoritaire en supprimant aléatoirement certaines de ses observations, de manière à équilibrer le jeu de données.

Algorithme

Soit $X_M \subset \mathbb{R}^p$ l'ensemble des observations appartenant à la classe majoritaire, avec n_M observations, et $X_m \subset \mathbb{R}^p$ la classe minoritaire, avec n_m observations.

L'objectif est de ramener la taille de la classe majoritaire à n_m, en sélectionnant aléatoirement un sous-ensemble $\tilde{X}_M \subset X_M$ tel que $|\tilde{X}_M| = n_m$.

1. Sélectionner aléatoirement n_m observations de X_M, sans remise :

$$\tilde{X}_M = \{x_1^*, \ldots, x_{n_m}^*\} \subset X_M$$

2. Construire le nouvel ensemble équilibré :

$$X^{\text{final}} = \tilde{X}_M \cup X_m, \quad \text{avec } |\tilde{X}_M| = |X_m| = n_m$$

Le RUS est souvent utilisé lorsqu'on dispose d'un grand volume de données majoritaires, et qu'on souhaite réduire les temps de calcul. Toutefois, en supprimant aléatoirement des observations, cette méthode peut entraîner une perte d'information importante, notamment si les données supprimées contiennent des structures représentatives essentielles.

3.5.5 Sous-échantillonnage avec la méthode Condensed Nearest Neighbors (ENN)

Dans le cas d'un problème de classification déséquilibré, la classe majoritaire peut dominer l'apprentissage du modèle, ce qui entraîne un biais et une mauvaise généralisation. Une solution consiste à appliquer une méthode de sous-échantillonnage (under-sampling) afin de réduire la taille de la classe majoritaire.

Le Condensed Nearest Neighbor (CNN) est une technique qui vise à identifier un sous-ensemble minimal d'exemples d'entraînement (prototype set) permettant de classifier correctement les autres observations à l'aide d'un algorithme des 1-plus proche voisin (1-NN). Contrairement à l'ENN qui supprime les observations ambiguës, le CNN conserve uniquement les observations les plus informatives pour réduire la redondance dans la classe majoritaire.

3.5.6 Formulation mathématique

Soit :

— X_M l'ensemble des observations appartenant à la classe majoritaire, avec n_M observations.
— X_m l'ensemble des observations appartenant à la classe minoritaire, avec n_m observations.
— $D = X_M \cup X_m$ l'ensemble des données d'apprentissage avant rééchantillonnage.

L'algorithme CNN suit les étapes suivantes :

1. Initialiser un ensemble S avec un exemple aléatoire de chaque classe :

$$S = \{s_m, s_M\}, \quad s_m \in X_m, \quad s_M \in X_M$$

2. Pour chaque observation $x_i \in D \setminus S$, vérifier si elle est correctement classée par un classifieur 1-NN entraîné uniquement sur S.
3. Si x_i est mal classée, l'ajouter à S :

$$S \leftarrow S \cup \{x_i\}, \quad \text{si} \quad \hat{y}_i \neq y_i$$

où $\hat{y}_i$ est la prédiction faite par 1-NN avec l'ensemble S.
4. Répéter l'étape 2 jusqu'à ce qu'aucune observation ne soit mal classée.

À la fin du processus, on obtient un ensemble condensé S, contenant un nombre réduit d'observations, principalement issues de la classe minoritaire et des frontières de décision, ce qui permet de rééquilibrer les données sans sacrifier la qualité de la classification.

3.5.7 Combinaison du sur-échantillonnage et du sous-échantillonnage : SMOTE + Tomek Links

L'approche hybride de rééchantillonnage vise à combiner les avantages du *over-sampling* et du under-sampling afin d'obtenir un jeu de données mieux équilibré, tout en réduisant le bruit et en conservant la structure des classes. L'une des techniques les plus utilisées est la combinaison de SMOTE avec les Tomek Links.

Algorithme

Soit :

- X_M l'ensemble des observations de la classe majoritaire, avec n_M observations.
- X_m l'ensemble des observations de la classe minoritaire, avec n_m observations.
- $D = X_M \cup X_m$ l'ensemble des données avant rééchantillonnage.

Le processus se déroule en deux étapes :

1. Génération de nouveaux points par SMOTE

Dans un premier temps, SMOTE est appliqué à la classe minoritaire afin de générer des observations synthétiques x_{new}. La génération suit la règle :

$$x_{\text{new}} = x_i + \lambda \cdot (x_{nn} - x_i), \quad \lambda \sim U(0,1)$$

où :

- $x_i \in X_m$ est une observation de la classe minoritaire.
- x_{nn} est un voisin de x_i parmi ses k plus proches voisins.
- λ est un facteur aléatoire tiré d'une distribution uniforme $U(0,1)$.

L'ensemble des données devient alors $D' = X_M \cup (X_m \cup X_{\text{new}})$.

2. Nettoyage des observations avec Tomek Links

Après le sur-échantillonnage, un sous-échantillonnage sélectif est appliqué avec la méthode Tomek Links, qui vise à supprimer les observations de la classe majoritaire qui forment des paires appelées **Tomek Links**.

Une paire de points (x_i, x_j) constitue un Tomek Link si :

$$d(x_i, x_j) = \min_{x_k \in D'} d(x_i, x_k)$$

où :

- $x_i \in X_M$ et $x_j \in X_m$ sont de classes différentes.
- x_j est le plus proche voisin de x_i et vice versa.

Ces paires sont supprimées pour améliorer la séparation des classes et éviter le chevauchement.

L'ensemble final après nettoyage est noté D''.

3.5.8 Combinaison du sur-échantillonnage et du sous-échantillonnage : SMOTE + ENN

La méthode SMOTEENN combine deux techniques complémentaires pour traiter le déséquilibre des classes : le sur-échantillonnage synthétique via SMOTE (Synthetic Minority Over-sampling Technique) et le sous-échantillonnage par édition via ENN (Edited Nearest Neighbours). L'objectif de cette méthode est de rééquilibrer le jeu de données tout en nettoyant le bruit introduit par des observations ambiguës ou mal classées.

SMOTEENN est donc une approche hybride :

— SMOTE génère de nouvelles instances synthétiques pour la classe minoritaire.
— ENN supprime des observations (principalement de la classe majoritaire) mal classées selon leurs k plus proches voisins.

Cette combinaison permet d'améliorer à la fois la représentation de la classe minoritaire et la qualité globale du jeu de données en réduisant les points de bruit ou ambigus proches de la frontière de décision.

Algorithme

Soit $X_m \subset \mathbb{R}^p$ la classe minoritaire et $X_M \subset \mathbb{R}^p$ la classe majoritaire.

1. **Sur-échantillonnage avec SMOTE** : Générer un ensemble X_m^{SMOTE} d'observations synthétiques selon la méthode SMOTE :

$$x_{\text{new}} = x_i + \lambda \cdot (x_{nn} - x_i), \quad \lambda \sim U(0,1)$$

   où $x_i \in X_m$, x_{nn} est un voisin minoritaire parmi les k plus proches, et λ est un facteur de pondération aléatoire.
2. **Fusion des données** : Construire un ensemble enrichi :

$$X^{\text{SMOTE}} = X_m \cup X_m^{\text{SMOTE}} \cup X_M$$

3. **Nettoyage avec ENN (Edited Nearest Neighbours)** : Pour chaque observation $x_i \in X^{\text{SMOTE}}$, identifier ses k plus proches voisins. Si la majorité des voisins n'appartient pas à la même classe que x_i, alors x_i est supprimé :

$$x_i \in X^{\text{SMOTE}} \text{ est retiré si majorité des voisins } \neq \text{classe de } x_i$$

4. Le jeu de données final devient :

$$X^{\text{final}} = \text{SMOTE}(X_m) \cup X_M - \text{ENN-corrected points}$$

3.6 Les modèles de Machine Learning

3.6.1 La régression logistique

La régression logistique est un modèle de classification utilisé pour prédire la probabilité d'appartenance d'une observation à une classe binaire. Contrairement à la régression linéaire, elle est spécifiquement conçue pour modéliser une variable cible discrète, en utilisant la fonction logistique (ou sigmoïde) pour contraindre la sortie entre 0 et 1.

Formulation mathématique

Soit $\mathbf{x} = (x_1, x_2, \ldots, x_p)^T \in \mathbb{R}^p$ un vecteur de variables explicatives, et $y \in \{0, 1\}$ la variable cible.

La probabilité que l'observation appartienne à la classe 1 est modélisée comme suit :

$$P(y = 1 \mid \mathbf{x}) = \pi(\mathbf{x}) = \frac{1}{1 + \exp(-\mathbf{x}^T \boldsymbol{\beta})}$$

où $\boldsymbol{\beta} = (\beta_0, \beta_1, \ldots, \beta_p)^T$ est le vecteur des coefficients du modèle (incluant l'interception β_0).

On peut réécrire cette relation sous forme du *logit*, c'est-à-dire le logarithme du rapport de chances :

$$\log\left(\frac{\pi(\mathbf{x})}{1 - \pi(\mathbf{x})}\right) = \mathbf{x}^T \boldsymbol{\beta}$$

Interprétation des coefficients

Chaque coefficient β_j représente l'effet marginal de la variable x_j sur le *log-odds* (logarithme du rapport de chances). Plus précisément :

$\exp(\beta_j)$ = le facteur multiplicatif sur les chances associé à une augmentation d'une unité de x_j

Fonction de coût et estimation

Les coefficients $\boldsymbol{\beta}$ sont estimés par la méthode du maximum de vraisemblance. La vraisemblance pour un échantillon de taille n est donnée par :

$$\mathcal{L}(\boldsymbol{\beta}) = \prod_{i=1}^{n} \pi(\mathbf{x}_i)^{y_i} \cdot (1 - \pi(\mathbf{x}_i))^{1-y_i}$$

Le logarithme de la vraisemblance (log-vraisemblance) est :

$$\ell(\boldsymbol{\beta}) = \sum_{i=1}^{n} [y_i \log(\pi(\mathbf{x}_i)) + (1 - y_i) \log(1 - \pi(\mathbf{x}_i))]$$

L'objectif est de maximiser $\ell(\boldsymbol{\beta})$, ou de manière équivalente, de minimiser la fonction de coût négative :

$$J(\boldsymbol{\beta}) = -\ell(\boldsymbol{\beta})$$

Classification finale

Une fois le modèle entraîné, la prédiction finale est obtenue via un seuil $\tau \in [0, 1]$. Typiquement, on classe une observation $\mathbf{x}$ comme appartenant à la classe 1 si :

$$\pi(\mathbf{x}) \geq \tau$$

avec $\tau = 0{,}5$ par défaut, sauf cas de classes déséquilibrées où un autre seuil peut être plus pertinent.

3.6.2 L'arbre de décision

L'arbre de décision est un modèle d'apprentissage supervisé non paramétrique utilisé à la fois pour la classification et la régression. Il repose sur une structure arborescente dans laquelle chaque nœud interne représente un test sur une variable, chaque branche un résultat possible du test, et chaque feuille une prédiction finale.

Dans le cas d'une classification binaire, l'arbre cherche à diviser l'espace des variables explicatives en sous-espaces homogènes selon la variable cible.

Principe de construction

Soit $\mathcal{D} = \{(\mathbf{x}_i, y_i)\}_{i=1}^n$, avec $\mathbf{x}_i \in \mathbb{R}^p$, un jeu de données où la variable cible $y_i \in \{0, 1\}$.

L'algorithme de construction repose sur une procédure récursive :

1. À chaque nœud, on cherche la variable x_j et le seuil s qui divisent $\mathcal{D}$ en deux sous-ensembles $\mathcal{D}_g$ et $\mathcal{D}_d$, tels que :
$$\mathcal{D}_g = \{(\mathbf{x}_i, y_i) \mid x_{ij} \leq s\}, \quad \mathcal{D}_d = \{(\mathbf{x}_i, y_i) \mid x_{ij} > s\}$$
2. Le meilleur split est celui qui maximise le gain d'information, mesuré par un critère d'impureté (voir ci-dessous).
3. Le processus se répète récursivement sur les sous-ensembles, jusqu'à un critère d'arrêt (profondeur maximale, taille minimale d'un nœud, pureté atteinte, etc.).

Critères d'impureté

Les deux critères les plus couramment utilisés pour mesurer l'impureté d'un nœud sont :

— **Indice de Gini** :
$$G(t) = 1 - \sum_{k=1}^{K} p_k^2$$
où p_k est la proportion d'observations de la classe k dans le nœud t.

— **Entropie (Information Gain)** :
$$H(t) = -\sum_{k=1}^{K} p_k \log_2(p_k)$$

— Le **gain d'information** est alors :
$$\Delta H = H(t) - \left(\frac{n_g}{n} H(t_g) + \frac{n_d}{n} H(t_d)\right)$$
où t_g et t_d sont les nœuds enfants gauche et droit.

Prédiction

Pour une nouvelle observation $\mathbf{x}_{\text{new}}$, le modèle suit les règles de partition depuis la racine jusqu'à une feuille, et renvoie la classe majoritaire dans cette feuille.

Élagage (Pruning)

Un arbre non limité peut sur-apprendre les données (overfitting). L'élagage vise à réduire la complexité de l'arbre en supprimant certaines branches peu utiles.

Deux stratégies principales existent :

— **Pré-élagage (pre-pruning)** : arrêter la construction en amont selon des critères (profondeur maximale, nombre minimal d'observations, etc.).
— **Post-élagage (post-pruning)** : construire un arbre complet puis retirer a posteriori certaines branches en utilisant un jeu de validation ou un critère de complexité (comme le coût-complexité).

3.6.3 La forêt aléatoire (Random Forest)

La forêt aléatoire (Random Forest) est un modèle d'apprentissage supervisé basé sur un ensemble d'arbres de décision. Elle améliore la robustesse et la performance d'un arbre unique en combinant plusieurs arbres construits sur des sous-échantillons aléatoires des données. C'est une méthode d'agrégation dite *bagging* (Bootstrap Aggregating).

Principe général

L'idée est de construire un ensemble de B arbres de décision $\{T_b\}_{b=1}^{B}$, chacun entraîné sur un échantillon bootstrap différent du jeu de données d'origine, puis d'agréger leurs prédictions.

— En régression : la prédiction finale est la moyenne des prédictions des arbres.
— En classification : la classe prédite est celle obtenant la majorité des votes (*mode*).

Algorithme

Pour un jeu de données $\mathcal{D} = \{(\mathbf{x}_i, y_i)\}_{i=1}^{n}$:

1. Pour chaque itération $b = 1, \ldots, B$:
   — Tirer un échantillon bootstrap $\mathcal{D}_b$ de taille n avec remise.
   — Construire un arbre de décision T_b, en sélectionnant aléatoirement m variables (avec $m < p$) à chaque nœud pour déterminer la meilleure séparation.
2. Pour une nouvelle observation $\mathbf{x}$, agréger les prédictions :

$$\hat{y} = \text{mode}\left(T_1(\mathbf{x}), T_2(\mathbf{x}), \ldots, T_B(\mathbf{x})\right) \quad \text{(classification)}$$

3.6.4 Extreme Gradient Boosting

XGBoost (Extreme Gradient Boosting) est une méthode d'ensemble basée sur le principe du *boosting* de gradient. Elle consiste à construire un ensemble de modèles faibles (typiquement des arbres de décision peu profonds) de manière séquentielle, en corrigeant à chaque étape les erreurs commises par le modèle précédent.

XGBoost améliore les méthodes de boosting traditionnelles (comme AdaBoost et Gradient Boosting Machines) en introduisant des techniques d'optimisation supplémentaires : régularisation, parallélisation, gestion des valeurs manquantes, et traitement efficace de grands volumes de données.

Algorithme

Soit un jeu de données d'entraînement $\mathcal{D} = \{(\mathbf{x}_i, y_i)\}_{i=1}^{n}$, avec $\mathbf{x}_i \in \mathbb{R}^p$ les variables explicatives, et $y_i \in \mathbb{R}$ la variable à prédire.

Le modèle XGBoost cherche à approximer la fonction de prédiction $\hat{y}_i = \phi(\mathbf{x}_i) = \sum_{k=1}^{K} f_k(\mathbf{x}_i)$, où chaque $f_k \in \mathcal{F}$ est un arbre de décision (structure de type CART : Classification And Regression Tree), et $\mathcal{F}$ est l'espace des arbres.

Le modèle est appris en minimisant la fonction objectif suivante :

$$\mathcal{L}(\phi) = \sum_{i=1}^{n} l(y_i, \hat{y}_i) + \sum_{k=1}^{K} \Omega(f_k)$$

où :

— $l(y_i, \hat{y}_i)$ est une fonction de perte (ex. : carré pour la régression, logistique pour la classification).
— $\Omega(f_k) = \gamma T + \frac{1}{2}\lambda \|w\|^2$ est un terme de régularisation qui pénalise la complexité de l'arbre, avec T le nombre de feuilles et w les scores associés aux feuilles.

Étapes de l'algorithme

À chaque itération t, on ajoute un nouvel arbre f_t pour minimiser la fonction objectif en utilisant un développement de Taylor d'ordre 2 :

$$\mathcal{L}^{(t)} \simeq \sum_{i=1}^{n} \left[g_i f_t(\mathbf{x}_i) + \frac{1}{2} h_i f_t^2(\mathbf{x}_i) \right] + \Omega(f_t)$$

où :

$$g_i = \frac{\partial l(y_i, \hat{y}_i^{(t-1)})}{\partial \hat{y}_i^{(t-1)}}, \quad h_i = \frac{\partial^2 l(y_i, \hat{y}_i^{(t-1)})}{\partial (\hat{y}_i^{(t-1)})^2}$$

Ces dérivées sont appelées respectivement gradients et hessiennes, et servent à guider la construction de l'arbre optimal à l'étape t.

3.6.5 Classification par Réseaux de Neurones (Neural Network Classifier)

Les réseaux de neurones artificiels sont des modèles d'apprentissage supervisé inspirés du fonctionnement du cerveau humain. Utilisés en classification, ils permettent de capturer des relations complexes non linéaires entre les variables d'entrée et la variable cible. Un réseau de neurones se compose de plusieurs couches de nœuds (ou "neurones"), connectées entre elles.

Structure d'un réseau de neurones

Un réseau de neurones simple (Perceptron Multi-Couche - MLP) comprend :

— Une couche d'entrée (input layer),
— Une ou plusieurs couches cachées (hidden layers),
— Une couche de sortie (output layer).

Chaque neurone réalise un calcul de la forme :

$$z_j = \sum_{i=1}^{p} w_{ij} x_i + b_j \quad \text{et} \quad a_j = \phi(z_j)$$

où :

- x_i : entrées du neurone,
- w_{ij} : poids synaptiques,
- b_j : biais,
- ϕ : fonction d'activation (non-linéaire),
- a_j : sortie du neurone.

Fonctions d'activation courantes

- **Sigmoïde** :

$$\phi(z) = \frac{1}{1 + e^{-z}}, \quad \text{sortie dans } (0, 1)$$

- **Tanh** :

$$\phi(z) = \tanh(z) = \frac{e^{z} - e^{-z}}{e^{z} + e^{-z}}, \quad \text{sortie dans } (-1, 1)$$

- **ReLU (Rectified Linear Unit)** :

$$\phi(z) = \max(0, z)$$

- **Softmax** (couche de sortie pour classification multi-classes) :

$$\phi(z_i) = \frac{e^{z_i}}{\sum_{j=1}^{K} e^{z_j}}, \quad i = 1, \ldots, K$$

Fonction de coût

Pour un problème de classification binaire, la fonction de perte est souvent la **log-loss** :

$$\mathcal{L} = -\frac{1}{n} \sum_{i=1}^{n} [y_i \log(\hat{y}_i) + (1 - y_i) \log(1 - \hat{y}_i)]$$

En multi-classes, on utilise l'entropie croisée (cross-entropy).

Apprentissage du réseau (Backpropagation)

L'apprentissage consiste à ajuster les poids pour minimiser la fonction de coût via l'algorithme de rétropropagation (*backpropagation*) combiné avec une méthode d'optimisation, comme la descente de gradient stochastique :

$$w_{ij}^{(t+1)} = w_{ij}^{(t)} - \eta \cdot \frac{\partial \mathcal{L}}{\partial w_{ij}}$$

où η est le taux d'apprentissage.

3.6.6 Critères de performance adaptés au déséquilibre des classes

Pour évaluer les performances d'un classifieur binaire, on s'appuie généralement sur la matrice de confusion (aussi appelée matrice d'erreur), qui résume le nombre d'instances correctement ou incorrectement classées par le modèle en fonction de leur classe d'appartenance réelle. Le tableau suivant illustre cette matrice :

Modalité observée / Modalité prédite	**C1**	**C0**	**Total**
C1	Vrais Positifs (VP)	Faux Négatifs (FN)	n_{C1}
C0	Faux Positifs (FP)	Vrais Négatifs (VN)	n_{C0}
Total	$\hat{n}_{C1}$	$\hat{n}_{C0}$	n

TABLE 3.1 – Matrice de confusion

Traditionnellement, on calcule à partir de cette matrice le **Taux de Bon Classement (TBC)** (ou accuracy) :

$$TBC = \frac{VP + VN}{VP + FN + FP + VN}$$

Cependant, dans des contextes de déséquilibre des classes, cette mesure n'est pas adaptée, car elle accorde le même poids aux erreurs des deux classes, masquant ainsi les performances sur la classe minoritaire, souvent celle qui nous intéresse le plus.

À partir de la matrice, on définit :

— **Rappel (Sensibilité, True Positive Rate)** :

$$TVP = \frac{VP}{VP + FN}$$

— **Spécificité (True Negative Rate)** :

$$TVN = \frac{VN}{FP + VN}$$

— **Taux de Faux Positifs (False Positive Rate)** :

$$TFP = \frac{FP}{FP + VN}$$

— **Taux de Faux Négatifs (False Negative Rate)** :

$$TFN = \frac{FN}{VP + FN}$$

Pour équilibrer les performances entre les deux classes, plusieurs mesures combinées ont été introduites.

— **AUC – Area Under the ROC Curve** :

$$AUC = \frac{1 + TVP - TFP}{2}$$

— **Courbe précision/rappel** avec la précision définie par :

$$Précision = \frac{VP}{VP + FP}$$

— **Balanced Accuracy (BAC)** :

$$BAC = \frac{TVP + TVN}{2}$$

— **G-Mean** :

$$G\text{-}mean = \sqrt{TVP \times TVN}$$

— **Adjusted G-Mean (AGM)** :

$$AGM = \begin{cases} GM + TVN \times \frac{FP+VN}{1+FP+VN}, & \text{si } TVP > 0 \\ 0, & \text{sinon} \end{cases}$$

— **G-Measure** :

$$G\text{-}measure = \sqrt{Recall \times Précision}$$

— **F-Measure (F_β)** :

$$F_\beta = (1 + \beta^2) \times \frac{Précision \times Recall}{\beta^2 \times Précision + Recall}$$

— **Index of Balanced Accuracy (IBA)** :

$$IBA_\alpha(C) = (1 + \alpha \times Dom) \times C, \quad \text{avec } Dom = TVP - TVN$$

Dans le cadre de ce travail, c'est la G-Mean qui a été retenue comme critère principal d'évaluation. Le G-mean, ou moyenne géométrique, constitue un indicateur essentiel dans l'évaluation des modèles appliqués aux jeux de données déséquilibrés (He et Garcia, 2009). Sa pertinence réside dans sa capacité à combiner sensibilité (rappel des classes minoritaires) et spécificité (rappel des classes majoritaires), offrant ainsi une mesure équilibrée des performances globales, particulièrement adaptée aux contextes où les classes minoritaires ont une importance égale aux classes majoritaires (Branco et al., 2016).

Chapitre 4

Présentation des résultats

***

Dans ce chapitre, il sera question de présenter les différents résultats des méthodes utilisées.

4.1 Les variables sélectionnées

Le tableau ci-dessous présente les variables explicatives sélectionnées . La colonne Total indique le nombre de fois qu'une variable a été retenue parmi les différentes approches, ce qui permet de repérer les variables les plus robustes et récurrentes, ceux qui ont été sélectionnées par au moins deux méthodes.

Feature	Pearson	Chi-2	RFE	Logistics	Random Forest	LightGBM	Total
Net_Sales	False	False	True	True	True	True	4
Agency_Type	False	True	True	True	False	True	4
Agency_C2B	True	True	True	True	False	False	4
Agency_LWC	True	False	True	True	False	False	3
SINGAPORE	False	True	False	False	False	True	2
Product_Name_Travel_Cruise_Protect	False	False	True	True	False	False	2
Product_Name_Cancellation_Plan	False	True	True	False	False	False	2
Product_Name_Basic_Plan	False	False	True	True	False	False	2
Gender	False	True	False	False	True	False	2
Duration	False	False	False	False	True	True	2
Distribution_Channel	False	False	True	True	False	False	2
Commission	False	False	False	False	True	True	2
Agency_RAB	True	False	False	True	False	False	2
Agency_JZI	False	False	True	True	False	False	2
Age	False	False	False	False	True	True	2

TABLE 4.1 – Variables explicatives sélectionnées selon différentes méthodes de sélection

4.2 Gestion du déséquilibre des données : Rééchantillonnage

Une difficulté majeure rencontrée dans la prédiction des sinistres en assurance voyage est le déséquilibre de la variable cible. Dans notre cas, la majorité des observations correspondent à l'absence de sinistre (*No*), tandis qu'une minorité seulement représente des sinistres déclarés (*Yes*).

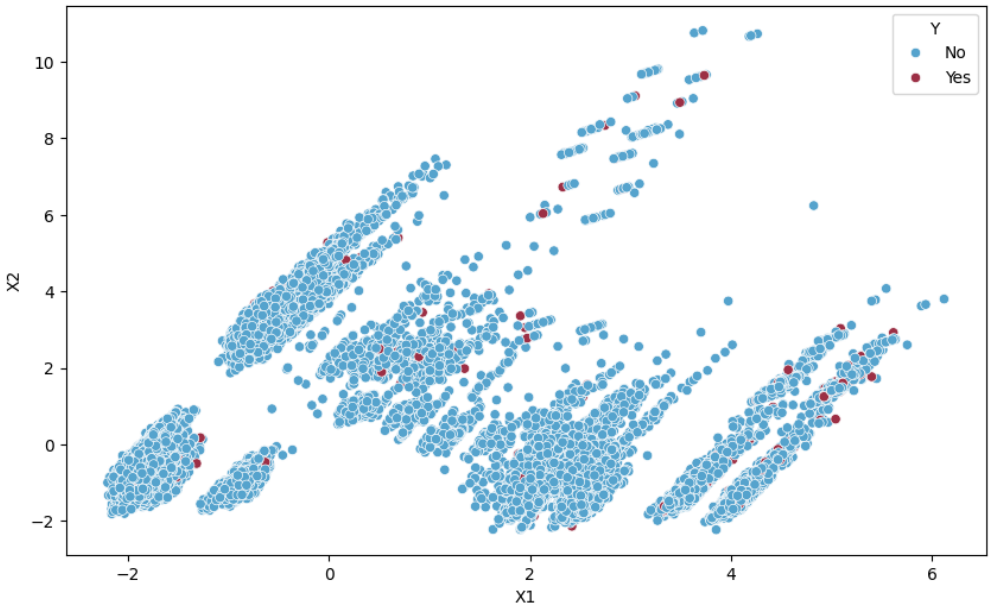

FIGURE 4.1 – No resampling échantillon

Pour corriger ce biais, plusieurs techniques de **rééchantillonnage** ont été appliquées et comparées :

Méthode	**Description**
Random OverSampling (ROS)	Réplication aléatoire des exemples minoritaires pour équilibrer les classes.
Random UnderSampling (RUS)	Réduction aléatoire des exemples majoritaires pour atteindre l'équilibre.
SMOTE	Création synthétique d'exemples minoritaires en interpolant entre voisins proches.
CNN (Condensed Nearest Neighbor)	Sélection des points les plus informatifs proches des frontières.
ADASYN	Génération adaptative d'exemples minoritaires dans les zones difficiles.
SMOTE + Tomek Links	Application de SMOTE suivie du nettoyage des frontières avec les paires Tomek.
SMOTE + ENN (Edited Nearest Neighbors)	Application de SMOTE suivie de la suppression des points ambigus avec ENN.

Les graphiques ci-dessous illustrent les distributions des classes après application de chaque technique :

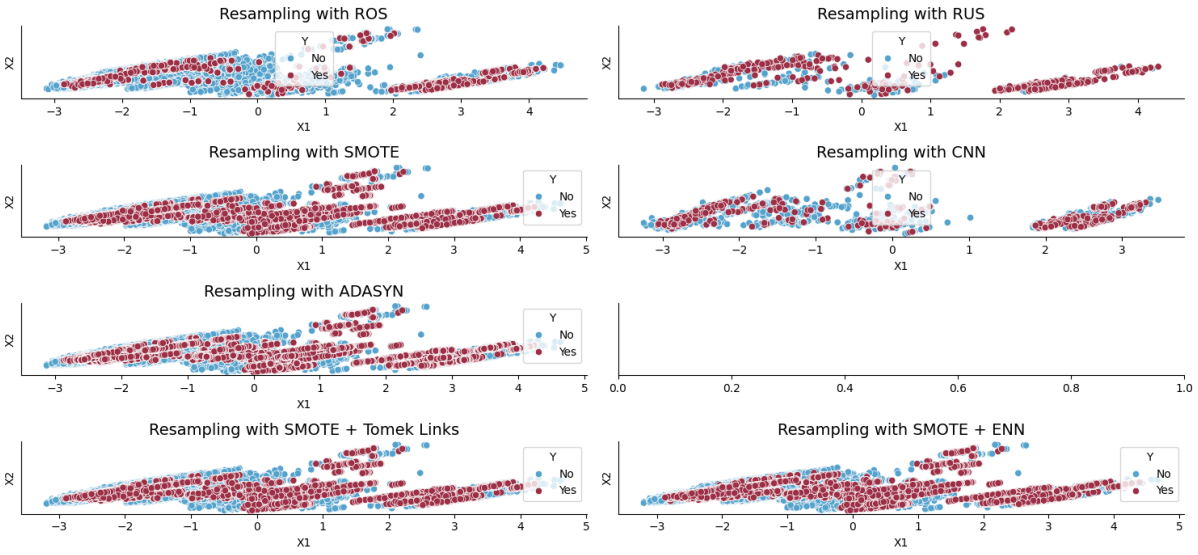

FIGURE 4.2 – Les échantillons Train

— Avec **ROS**, les observations minoritaires sont simplement dupliquées, ce qui densifie leur présence mais sans ajouter de diversité, risquant de conduire à un surapprentissage.
— Avec **RUS**, la taille de la majorité est drastiquement réduite, simplifiant la frontière de décision mais au prix d'une perte d'information importante.
— Les techniques synthétiques comme **SMOTE** et **ADASYN** génèrent de nouveaux points minoritaires dans l'espace des caractéristiques, permettant de combler les zones où les minoritaires étaient sous-représentés. Cela peut améliorer l'apprentissage, mais introduit aussi le risque de créer des points artificiels moins représentatifs.
— Les techniques combinées, notamment **SMOTE + Tomek Links** et **SMOTE + ENN**, apportent un nettoyage supplémentaire en retirant les points proches des frontières qui sont susceptibles d'être du bruit ou des ambiguïtés. Cela permet d'améliorer la qualité des ensembles équilibrés.
— Enfin, **CNN** sélectionne uniquement les points essentiels de la majorité, réduisant l'ensemble d'apprentissage pour se concentrer sur les zones critiques.

Pour chaque méthode, des modèles prédictifs seront entraînés et comparés en utilisant des métriques adaptées aux problèmes déséquilibrés, notamment la G-Mean. Ces comparaisons permettront de déterminer quelle approche de rééchantillonnage et quel modèle maximisent la capacité du à identifier les sinistres tout en minimisant les faux positifs.

4.3 Comparaison des modèles

L'évaluation des modèles selon le G-Mean révèle plusieurs constats :

Le Random Forest démontre une supériorité marquée parmi l'ensemble des algorithmes évalués. Ce modèle maintient des scores G-mean supérieurs à 21% pour presque toutes les techniques de rééchantillonnage (ADASYN, CNN, RoS, RuS, SMOTE, SMOTE+ENN, SMOTE+TOMEK), atteignant un pic à 22% avec la méthode CNN.

La régression logistique présente également des valeurs G-mean oscillant entre 20% et 21% selon les méthodes employées. Les réseaux neuronaux affichent des performances moins bonnes que les modèles ci-dessus, généralement comprises entre 17% et 19%. Bien qu'inférieurs au Random Forest, ils surpassent néanmoins les arbres de décision simples, témoignant de leur capacité à appréhender des structures plus complexes dans les données (LeCun et al., 2015).

En revanche, les résultats obtenus avec les modèles Decision Tree et XGBoost s'avèrent moins satisfaisants. Le Decision Tree utilisé seul obtient des scores modestes, généralement entre 12% et 13% de G-mean, suggérant un probable surapprentissage sur les classes majoritaires, phénomène déjà identifié par Quinlan (1986) et plus récemment par Lemaître et al. (2017). Plus surprenant encore, XGBoost présente des performances relativement faibles, parfois à peine supérieures à 11, contredisant les résultats obtenus par Chen et Guestrin (2016) dans d'autres contextes.

L'analyse comparative des techniques de rééchantillonnage révèle que CNN (Cluster-based under-sampling) génère les résultats les plus faibles, voire nuls, pour la majorité des modèles. À l'inverse, les méthodes telles qu'ADASYN, Random OverSampling (RoS), Random UnderSampling (RuS), SMOTE, SMOTE+ENN et SMOTE+TOMEK améliorent globalement les performances, particulièrement pour le Random Forest et la régression logistique. Les approches RoS et RuS semblent légèrement se distinguer comme les techniques les plus équilibrées, engendrant des gains de performance plus uniformes à travers l'ensemble des modèles.

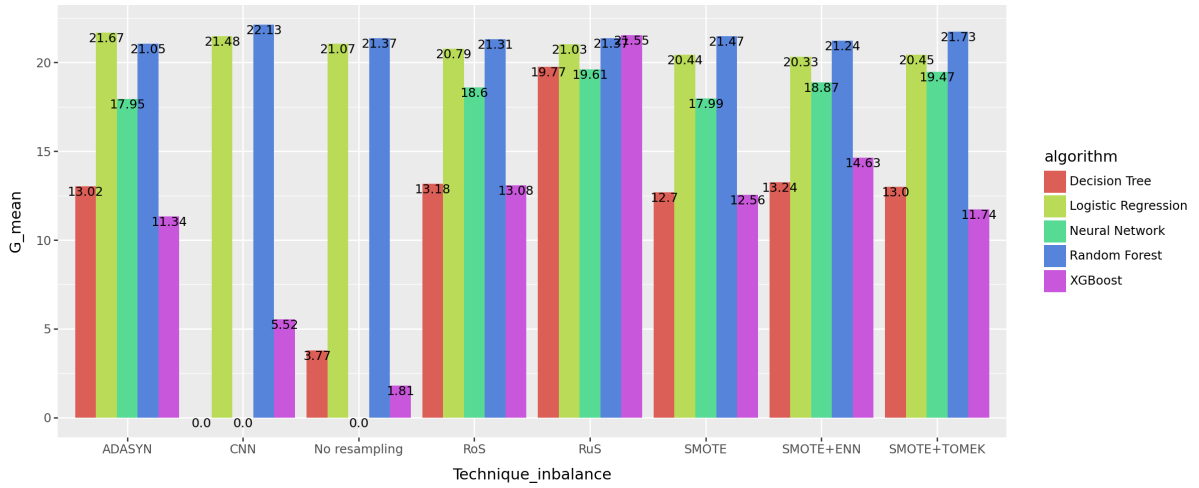

FIGURE 4.3 – Résutat des modèles selon le Gmean

L'analyse selon l'AUC montre que le modèle Random Forest se distingue encore une fois comme l'un des plus performants, affichant des AUC systématiquement supérieurs à 74%, et atteignant jusqu'à 76% selon les techniques (notamment ADASYN, SMOTE+ENN, SMOTE+TOMEK). Cette constance confirme sa robustesse, déjà observée avec le G-mean, et sa capacité à capter des relations complexes même sans rééchantillonnage.

La régression logistique obtient également de bonnes performances, avec des AUC allant de 74% à 75%, confirmant que ce modèle simple mais puissant reste compétitif lorsqu'il est bien calibré. Le réseau de neurones suit de près, avec des AUC oscillant autour de 72% à 73%.

Les arbres de décision simples (Decision Tree) présentent des résultats plus modestes,

avec des AUC souvent autour de 62% à 63%, sauf avec CNN et sans rééchantillonnage, où les performances chutent à environ 50%, indiquant une incapacité à mieux que le hasard. XGBoost affiche des résultats très moyens, avec des AUC autour de 58% à 59%, sauf sans rééchantillonnage ou avec CNN où il tombe également à 50%.

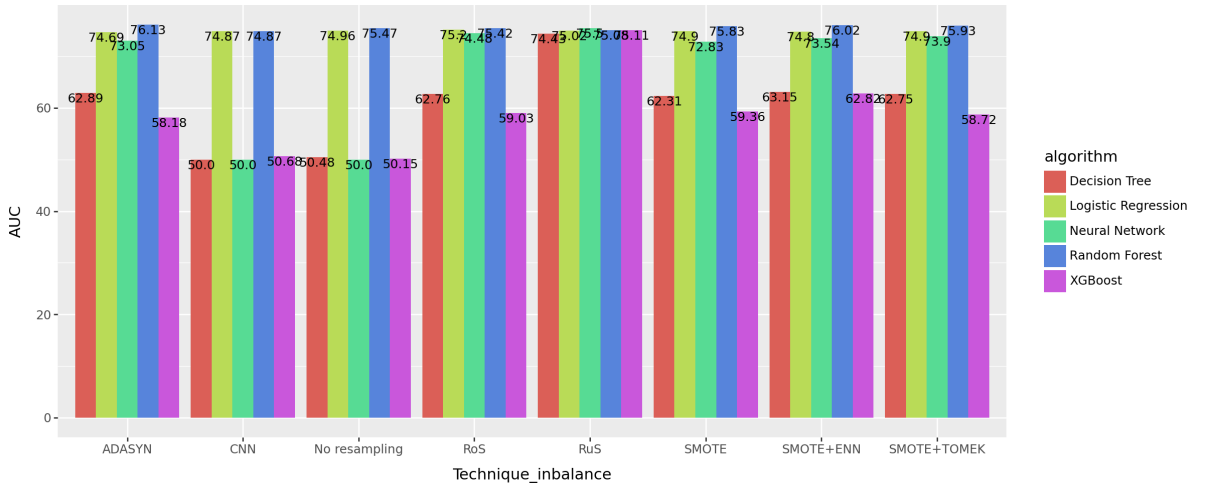

FIGURE 4.4 – Résutat des modèles selon l'AUC-ROC

4.4 Une approche "Algorithm-level" pour déterminer le seuil optimal

Après avoir identifié Random Forest associé au CNN comme la meilleure combinaison selon les métriques précédentes, une méthode algorithm-level a été appliquée pour améliorer encore la détection de la classe positive (Sinistre). Plutôt que de simplement rééchantillonner les données, cette approche consiste à ajuster le seuil de probabilité à partir duquel un individu est classé comme étant sinistre.

En pratique, le modèle Random Forest retourne pour chaque observation une probabilité d'appartenance à la classe positive (Yes). Par défaut, le seuil est fixé à 0.5 : au-dessus, on prédit la classe positive ; en dessous, la classe négative. Cependant, ce seuil par défaut n'est pas forcément optimal, surtout dans des contextes déséquilibrés où l'on souhaite maximiser la capacité à identifier correctement les cas positifs (True Positives) sans trop augmenter les faux positifs (False Positives).

Pour identifier le seuil optimal, la courbe ROC (Receiver Operating Characteristic) a été analysée. Cette courbe trace le taux de vrais positifs (TPR) en fonction du taux de faux positifs (FPR) pour différents seuils. À partir de ces points, le G-mean a été calculé pour chaque seuil. Celui qui maximise ce G-mean est considéré comme optimal, car il offre le meilleur compromis entre les deux types d'erreur.

Dans ce cas, le seuil optimal trouvé est de 0.44, avec un G-mean de 75.68%, un TPR (taux de vrais positifs) de 73.19% et un FPR (taux de faux positifs) de %21.74. Une visualisation de la courbe ROC a permis de localiser ce point optimal et de mieux comprendre comment l'ajustement du seuil améliore la performance du modèle.

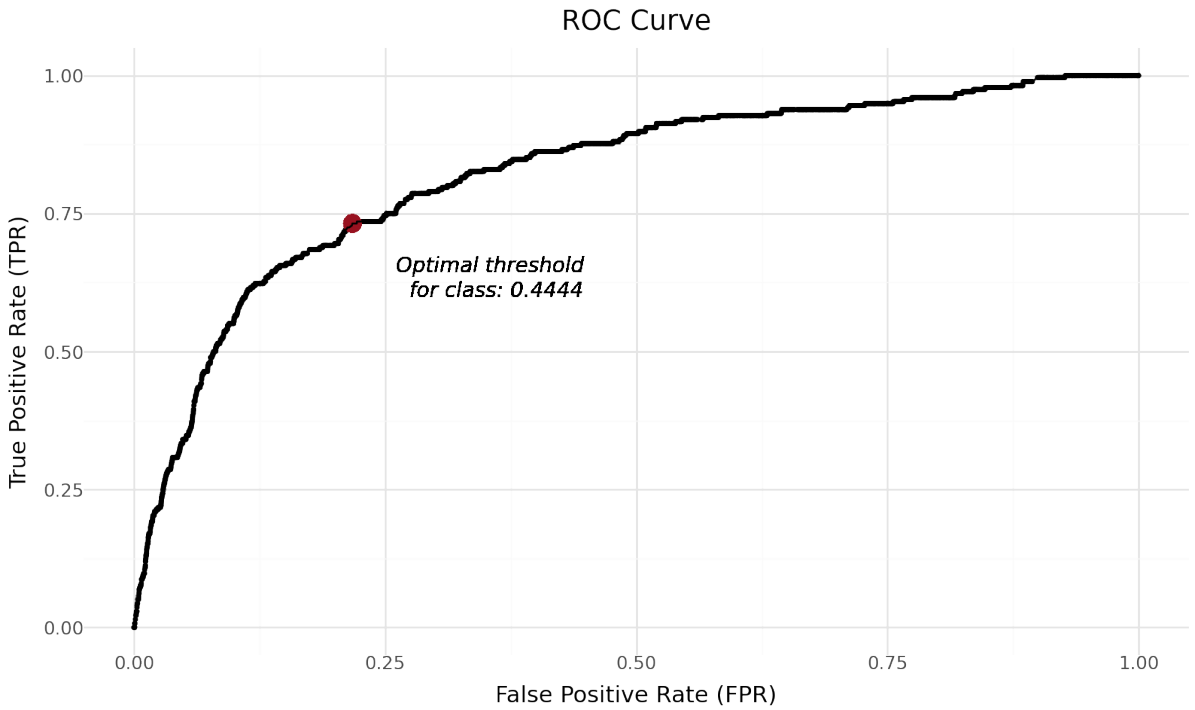

FIGURE 4.5 – Seuil optimal

Conclusion

***

Ce travail a mis en lumière l'importance cruciale de la prédiction des sinistres dans le domaine de l'assurance voyage, en se focalisant sur la problématique majeure du déséquilibre des classes. L'analyse approfondie menée sur les différentes approches de résolution de ce problème a permis d'identifier des combinaisons optimales de techniques pour améliorer significativement la performance prédictive des modèles.

Les résultats obtenus démontrent que l'algorithme Random Forest associé à la méthode de rééchantillonnage CNN (Cluster-based under-sampling) offre les meilleures performances selon la métrique G-mean, atteignant un score de 22%. Cette combinaison se distingue également par sa robustesse en termes d'AUC, avec des valeurs systématiquement supérieures à 74%. L'approche "algorithm-level" consistant à optimiser le seuil de décision a par ailleurs permis d'améliorer considérablement ces performances, atteignant un G-mean de 75.68% avec un seuil optimal de 0.44, offrant ainsi un compromis idéal entre la détection des sinistres (TPR de 73.19%) et la limitation des fausses alertes (FPR de 21.74%).

Ces résultats confirment l'hypothèse initiale selon laquelle la combinaison judicieuse de méthodes "data-level" et "algorithm-level" permet de surmonter efficacement le défi du déséquilibre des classes. Ils soulignent également l'importance de choisir des métriques d'évaluation appropriées, au-delà de la simple exactitude, pour mesurer la performance réelle des modèles dans ce contexte particulier. Pour les professionnels de l'assurance voyage, ces avancées offrent des perspectives concrètes d'optimisation de la gestion des risques. La capacité à mieux anticiper les sinistres permet non seulement d'affiner la tarification des polices, mais aussi d'améliorer la rentabilité globale tout en proposant des produits plus adaptés aux profils spécifiques des voyageurs. De plus, la méthodologie développée pourrait être étendue à d'autres branches de l'assurance IARD confrontées à des problématiques similaires de déséquilibre des classes.

Limites et recommandations

***

Malgré les résultats intéressants obtenus, ce mémoire présente plusieurs limites qu'il convient de souligner.

Premièrement, la qualité et la disponibilité des données ont constitué un frein notable. Les jeux de données utilisés, bien qu'exploitables, comportaient des lacunes (valeurs manquantes, variables peu renseignées, absence de certains détails contextuels) qui ont pu affecter la précision des modèles. Une collecte de données plus exhaustive, incluant des variables additionnelles comme le comportement de souscription, les antécédents de réclamation ou les conditions particulières des destinations, aurait sans doute permis de raffiner les analyses.

Deuxièmement, les modèles mis en œuvre restent limités dans leur capacité à capturer certaines dynamiques complexes, notamment les interactions non linéaires entre variables ou l'évolution temporelle des comportements assurantiels. Bien que des algorithmes avancés aient été testés, leur paramétrage et leur interprétation restent des défis, en particulier dans un cadre assurantiel où l'explicabilité est essentielle.

Troisièmement, l'évaluation des performances prédictives s'est principalement fondée sur des métriques standards (précision, rappel, courbe ROC), sans prise en compte explicite des impacts économiques (par exemple, le coût des fausses alertes ou des sinistres non prédits) ni des contraintes opérationnelles des compagnies d'assurance.

Au regard de ces limites, plusieurs recommandations peuvent être formulées. Il serait pertinent, dans de futurs travaux, de :

- Renforcer la qualité des données en collaborant étroitement avec les services de collecte et de gestion des contrats pour disposer de bases plus riches et mieux structurées.
- Explorer des modèles plus sophistiqués comme les approches par réseaux neuronaux profonds, les modèles séquentiels (type LSTM) ou les techniques d'ensemblage plus avancées, tout en veillant à préserver l'interprétabilité nécessaire à une application concrète.
- Intégrer une évaluation économique des performances, en quantifiant les gains ou pertes associés aux prédictions pour mieux aligner les résultats des modèles avec les objectifs stratégiques des assureurs.
-

Ces recommandations visent à renforcer la robustesse, la pertinence et l'impact des

travaux futurs, en contribuant à développer des outils prédictifs mieux adaptés aux besoins spécifiques du secteur de l'assurance voyage.

Références

***

Adil, M., Ansari, M. F., Alahmadi, A., Wu, J.-Z., & Chakrabortty, R. K. (2021). Solving the problem of class imbalance in the prediction of hotel cancelations : A hybridized machine learning approach. Processes, 9(10), 1713. https ://www.mdpi.com/2227-9717/9/10/1713

Adil, M., Wu, J.-Z., Chakrabortty, R. K., Alahmadi, A., Ansari, M. F., & Ryan, M. J. (2021). Attention-based STL-BiLSTM network to forecast tourist arrival. Processes, 9(10), 1759. https ://www.mdpi.com/2227-9717/9/10/1759

Aish, M. A. (2024). Predictive Modeling of Cerebral Strokes : An ADASYN-RF Approach for Imbalanced Data. VFAST Transactions on Software Engineering, 12(4), 12 26. https ://www.vfast.org/journals/index.php/VTSE/article/view/1932

Alsmariy, R., Healy, G., & Abdelhafez, H. (2020). Predicting cervical cancer using machine learning methods. International Journal of Advanced Computer Science and Applications, 11(7). https ://www.academia.edu/download/64997109/Paper$_2$3$_P$redicting$_C$ervical$_C$ancer$_u$sing$_M$achine

Attaran, M., & Deb, P. (2018). Machine learning : The new « big thing » for competitive advantage. International Journal of Knowledge Engineering and Data Mining, 5(4), 277. https ://doi.org/10.1504/IJKEDM.2018.095523

Batista, G. E. A. P. A., Prati, R. C., & Monard, M. C. (2004). A study of the behavior of several methods for balancing machine learning training data. ACM SIGKDD Explorations Newsletter, 6(1), 20 29. https ://doi.org/10.1145/1007730.1007735

Bian, Y., Yang, C., Zhao, J. L., & Liang, L. (2018). Good drivers pay less : A study of usage-based vehicle insurance models. Transportation research part A : policy and practice, 107, 20 34. https ://www.sciencedirect.com/science/article/pii/S096585641730561X

Bonde, L., & Bichanga, A. K. (2025). Improving Credit Card Fraud Detection with Ensemble Deep Learning-Based Models : A Hybrid Approach Using SMOTE-ENN. Journal of Computing Theories and Applications, 2(3), 384. https ://www.researchgate.net/profile/Lossan-

Bonde/publication/388908232$_I$mproving$_C$redit$_C$ard$_F$raud$_D$etection$_w$ith$_E$nsemble$_D$eep$_L$earning–Based$_M$odels$_A$Hybrid$_A$pproach$_U$sing$_S$MOTE–$_E$NN/links/67ac59b78311ce680c5e91f4/Improving–Credit – Card – Fraud – Detection – with – Ensemble – Deep – Learning – Based – Models – A – Hybrid – Approach – Using – SMOTE – ENN.pdf

Branco, P., Torgo, L., & Ribeiro, R. P. (2017). A Survey of Predictive Modeling on Imbalanced Domains. ACM Computing Surveys, 49(2), 1 50. https ://doi.org/10.1145/2907070
Brandt, J., & Lanzén, E. (2021). A comparative review of SMOTE and ADASYN in imbalanced data classification. https ://www.diva-portal.org/smash/record.jsf ?pid=diva2 :1519153

Chandola, V., Banerjee, A., & Kumar, V. (2009). Anomaly detection : A survey. ACM Computing Surveys, 41(3), 1 58. https ://doi.org/10.1145/1541880.1541882

Chawla, N. V., Bowyer, K. W., Hall, L. O., & Kegelmeyer, W. P. (2002). SMOTE : Synthetic minority over-sampling technique. Journal of artificial intelligence research, 16, 321 357. http ://www.jair.org/index.php/jair/article/view/10302

Chen, C., Liaw, A., & Breiman, L. (2004). Using random forest to learn imbalanced data. Statistics Department of University of California at Berkeley. Berkeley. Technical Report 666.

Chen, R.-C., Dewi, C., Huang, S.-W., & Caraka, R. E. (2020). Selecting critical features for data classification based on machine learning methods. Journal of Big Data, 7(1), 52. https ://doi.org/10.1186/s40537-020-00327-4

Cui, S., Wang, D., Wang, Y., Yu, P.-W., & Jin, Y. (2018). An improved support vector machine-based diabetic readmission prediction. Computer methods and programs in biomedicine, 166, 123 135. https ://www.sciencedirect.com/science/article/pii/S0169260718308083

Di Franco, G., & Santurro, M. (2021). Machine learning, artificial neural networks and social research. Quality & Quantity, 55(3), 1007 1025. https ://doi.org/10.1007/s11135-020-01037-y

Dimiduk, D. M., Holm, E. A., & Niezgoda, S. R. (2018). Perspectives on the Impact of Machine Learning, Deep Learning, and Artificial Intelligence on Materials, Processes, and Structures Engineering. Integrating Materials and Manufacturing Innovation, 7(3), 157 172. https ://doi.org/10.1007/s40192-018-0117-8

Dong, G., & Bailey, J. (2012). Contrast data mining : Concepts, algorithms, and applications. CRC Press. https ://books.google.com/books ?hl=fr&lr=&id=$_u xNRbzNdfAC$&$oi = fnd$&$pg = PP1$&$dq = +\%5B14\%5D + Dong + G + and + Bailey + J + 2012 + Contrast + data + mining : +concepts, +algorithms, +and + applications + (CRC + +Press)$&$ots = Pi6mV7skR3$&$sig = tJcdv8eaEdX6jJ4APRVIe5nTdhs$

Drummond, C., & Holte, R. C. (2003). C4. 5, class imbalance, and cost sensitivity : Why under-sampling beats over-sampling. Workshop on learning from imbalanced datasets II, 11(1–8). http ://www.eiti.uottawa.ca/ nat/Workshop2003/drummondc.pdf

ElSeddawy, A. I., Karim, F. K., Hussein, A. M., & Khafaga, D. S. (2022). Predictive Analysis of Diabetes-Risk with Class Imbalance. Computational Intelligence and Neuroscience, 2022, 1 16. https ://doi.org/10.1155/2022/3078025

Fitria, D., Saragih, T. H., Kartini, D., & Indriani, F. (2024). A Classification of Appendicitis Disease in Children Using SVM with KNN Imputation and SMOTE Approach. Journal of Electronics, Electromedical Engineering, and Medical Informatics, 6(3), 302 311. http ://jeeemi.org/index.php/jeeemi/article/view/470

Gong, J., & Kim, H. (2017). RHSBoost : Improving classification performance in imbalance data. Computational Statistics & Data Analysis, 111, 1 13. https ://www.sciencedirect.com/science/article

Hairani, H., & Priyanto, D. (2023). A new approach of hybrid sampling SMOTE and ENN to the accuracy of machine learning methods on unbalanced diabetes disease data. International Journal of Advanced Computer Science and Applications, 14(8). https ://www.researchgate.net/profile/Hairani-

Hairani/publication/373635667$_{A}N ew_{A}pproach_{o}f_{H}ybrid_{S}amplings_{S}MOTE_{a}nd_{E}NN_{t}o_{t}he_{A}ccuracy_{o}f_{M}ach$
$New-Approach-of-Hybrid-Sampling-SMOTE-and-ENN-to-the-Accuracy-of-Machine-Learning-Methods-on-Unbalanced-Diabetes-Disease-Data.pdf$

Haixiang, G., Yijing, L., Shang, J., Mingyun, G., Yuanyue, H., & Bing, G. (2017). Learning from class-imbalanced data : Review of methods and applications. Expert systems with applications, 73, 220 239.

https ://www.sciencedirect.com/science/article/pii/S0957417416307175 He, H., & Garcia, E. A. (2009). Learning from imbalanced data. IEEE Transactions on knowledge and data engineering, 21(9), 1263 1284.

https ://ieeexplore.ieee.org/abstract/document/5128907/

Kaur, H., Pannu, H. S., & Malhi, A. K. (2020). A Systematic Review on Imbalanced Data Challenges in Machine Learning : Applications and Solutions. ACM Computing Surveys, 52(4), 1 36.

https ://doi.org/10.1145/3343440

Khalili-Damghani, K., Abdi, F., & Abolmakarem, S. (2019). Solving customer insurance coverage recommendation problem using a two-stage clustering-classification model. International Journal of Management Science and Engineering Management, 14(1), 9 19. https ://doi.org/10.1080/17509653.2018.1467801

Kim, S.-Y., Jung, T.-S., Suh, E.-H., & Hwang, H.-S. (2006). Customer segmentation and strategy development based on customer lifetime value : A case study. Expert systems with applications, 31(1), 101 107. https ://www.sciencedirect.com/science/article/pii/S0957417405001934

Kumari, M., & Subbarao, N. (2022). A Hybrid Resampling Algorithms SMOTE and ENN Based Deep Learning Models for Identification of Marburg Virus Inhibitors. Future Medicinal Chemistry, 14(10), 701 715. https ://doi.org/10.4155/fmc-2021-0290

Kuo, R. J., Lin, S. Y., & Shih, C. W. (2007). Mining association rules through integration of clustering analysis and ant colony system for health insurance database in Taiwan. Expert Systems with Applications, 33(3), 794 808. https ://www.sciencedirect.com/science/article/pii/S09574174

Li, J., & Fong, S. (2016). Solving imbalanced dataset problems for high-dimensional image processing by swarm optimization. In Bio-Inspired Computation and Applications in Image Processing (p. 311 321). Elsevier. https ://www.sciencedirect.com/science/article/pii/B9780128045367000

Longadge, R., & Dongre, S. (2013). Class Imbalance Problem in Data Mining Review (arXiv :1305.1707). arXiv. https ://doi.org/10.48550/arXiv.1305.1707

Luque, A., Carrasco, A., Martín, A., & de Las Heras, A. (2019). The impact of class imbalance in classification performance metrics based on the binary confusion matrix. Pattern Recognition, 91, 216 231. https ://www.sciencedirect.com/science/article/pii/S0031320319300950

Mohammed, R., Rawashdeh, J., & Abdullah, M. (2020). Machine learning with oversampling and undersampling techniques : Overview study and experimental results. 2020 11th

international conference on information and communication systems (ICICS), 243 248. https ://ieeexplore.ieee.org/abstract/document/9078901/

More, A. S., & Rana, D. P. (2017). Review of random forest classification techniques to resolve data imbalance. 2017 1st International conference on intelligent systems and information management (ICISIM), 72 78. https ://ieeexplore.ieee.org/abstract/document/8122151/

Munshi, R. M. (2024). Novel ensemble learning approach with SVM-imputed ADASYN features for enhanced cervical cancer prediction. Plos one, 19(1), e0296107. https ://journals.plos.org/plosone/article ?id=10.1371/journal.pone.0296107

Muraru, M. M., Simó, Z., & Iantovics, L. B. (2024). Cervical Cancer Prediction Based on Imbalanced Data Using Machine Learning Algorithms with a Variety of Sampling Methods. Applied Sciences, 14(22), 10085. https ://www.mdpi.com/2076-3417/14/22/10085

Neumann, Ł., Nowak, R. M., Okuniewski, R., & Wawrzyński, P. (2019). Machine Learning-Based Predictions of Customers' Decisions in Car Insurance. Applied Artificial Intelligence, 33(9), 817 828. https ://doi.org/10.1080/08839514.2019.1630151

Oussous, A., Benjelloun, F.-Z., Lahcen, A. A., & Belfkih, S. (2018). Big Data technologies : A survey. Journal of King Saud University-Computer and Information Sciences, 30(4), 431 448. https ://www.sciencedirect.com/science/article/pii/S1319157817300034

Pesantez-Narvaez, J., Guillen, M., & Alcañiz, M. (2019). Predicting motor insurance claims using telematics data—XGBoost versus logistic regression. Risks, 7(2), 70. https ://www.mdpi.com/2227-9091/7/2/70

Phan-Mai, T.-A., Thai, T. T., Mai, T. Q., Vu, K. A., Mai, C. C., & Nguyen, D. A. (2023). Validity of Machine Learning in Detecting Complicated Appendicitis in a Resource-Limited Setting : Findings from Vietnam. BioMed Research International, 2023(1), 5013812. https ://doi.org/10.1155/2023/5013812

Putra, L. G. R., Marzuki, K., & Hairani, H. (2023). Correlation-based feature selection and Smote-Tomek Link to improve the performance of machine learning methods on cancer disease prediction. Engineering and Applied Science Research, 50(6), 577 583. https ://ph01.tci-thaijo.org/index.php/easr/article/view/253528

Quinlan, S., Afli, H., & O'Reilly, R. (2019). A Comparative Analysis of Classification Techniques for Cervical Cancer Utilising At Risk Factors and Screening Test Results. AICS, 400 411. https ://www.academia.edu/download/92068376/aics$_3$7.*pdf*

Schmidt, J., Marques, M. R., Botti, S., & Marques, M. A. (2019). Recent advances and applications of machine learning in solid-state materials science. npj computational materials, 5(1), 83. https ://www.nature.com/articles/s41524-019-0221-0

Siddappa, N. G., & Kampalappa, T. (2019). Adaptive condensed nearest neighbor for imbalance data classification. International Journal of Intelligent Engineering and Systems, 12(2), 104 113. https ://inass.org/2019/2019043011.pdf

Sisodia, D. S., Reddy, N. K., & Bhandari, S. (2017). Performance evaluation of class balancing techniques for credit card fraud detection. 2017 IEEE International Conference on power, control, signals and instrumentation engineering (ICPCSI), 2747 2752. https ://ieeexplore.ieee.org/abstract/document/8392219/

Somasundaram, A., & Reddy, U. S. (2016). Data imbalance : Effects and solutions for classification of large and highly imbalanced data. international conference on research in engineering, computers and technology (ICRECT 2016), 1 16. https ://www.researchgate.net/profile/Akila-

Somasundaram/publication/320895020$_{D}ata_{I}mbalance_{E}ffects_{a}nd_{S}olutions_{f}or_{C}lassification_{o}f_{L}arge_{a}n$
$Imbalance - Effects - and - Solutions - for - Classification - of - Large - and - Highly - Imbalanced - Data.pdfSui, Y., Wei, Y., \& Zhao, D.(2015).Computer - AidedLungNoduleRecognitionbySVMClassifierBasedonCombinationofRandom$

Undersampling and SMOTE. Computational and Mathematical Methods in Medicine, 2015, 1 13. https ://doi.org/10.1155/2015/368674

Tanimu, J. J., Hamada, M., Hassan, M., Kakudi, H., & Abiodun, J. O. (2022). A machine learning method for classification of cervical cancer. Electronics, 11(3), 463. https ://www.mdpi.com/2079-9292/11/3/463

Tarawneh, A. S., Hassanat, A. B., Altarawneh, G. A., & Almuhaimeed, A. (2022). Stop oversampling for class imbalance learning : A review. IEEe Access, 10, 47643 47660. https ://ieeexplore.ieee.org/abstract/document/9761871/

Thakur, S. S., & Singh, J. K. (2014). Prediction of Online Vehicle Insurance System using Decision Tree Classifier and Bayes Classifier–A Comparative Analysis. Int. J. Comput. Appl, 975, 8887. https ://citeseerx.ist.psu.edu/document ?repid=rep1&type=pdf&doi=bcadb01a80593ebb604bb72c9ddad6dee94b564b

Weiss, G. M., & Tian, Y. (2008). Maximizing classifier utility when there are data acquisition and modeling costs. Data Mining and Knowledge Discovery, 17(2), 253 282. https ://doi.org/10.1007/s10618-007-0082-x

Wongvorachan, T., He, S., & Bulut, O. (2023). A comparison of undersampling, oversampling, and SMOTE methods for dealing with imbalanced classification in educational data mining. Information, 14(1), 54. https ://www.mdpi.com/2078-2489/14/1/54

Wu, C.-H., Kao, S.-C., Su, Y.-Y., & Wu, C.-C. (2005). Targeting customers via discovery knowledge for the insurance industry. Expert Systems with Applications, 29(2), 291 299. https ://www.sciencedirect.com/science/article/pii/S0957417405000552

Xu, Z., Shen, D., Nie, T., & Kou, Y. (2020). A hybrid sampling algorithm combining M-SMOTE and ENN based on Random forest for medical imbalanced data. Journal of Biomedical Informatics, 107, 103465. https ://www.sciencedirect.com/science/article/pii/S1532046420300940

Yu, K., Ding, W., Simovici, D. A., & Wu, X. (2012). Mining emerging patterns by streaming feature selection. Proceedings of the 18th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, 60 68. https ://doi.org/10.1145/2339530.2339544

Annexes

***

4.5 Annexe 1 : Etapes de mise en place d'un modèle de machine Learning

Le processus de création d'un modèle d'apprentissage automatique comprend six étapes :
1. l'identification des besoins

2. la collecte et la préparation des données

3. la configuration du modèle

4. l'entraînement du modèle

5. l'évaluation des performances

6. le test et l'ajustement du modèle.

L'identification des besoins Avant de commencer à travailler sur un modèle d'apprentissage automatique, il est important de comprendre le problème en jeu. Pour ce faire, il est nécessaire d'identifier les besoins et les objectifs de l'entreprise afin de décider si l'on peut recourir à un problème supervisé ou non supervisé. En outre, il est également important de se familiariser avec les données que l'on dispose, d'identifier avec précision les résultats que l'on souhaite obtenir pour détecter au mieux les problèmes à résoudre.

La collecte et la préparation des données Une fois que les besoins et les objectifs sont bien identifiés à la première étape, il faudra identifier les données pertinentes pouvant permettre d'adresser efficacement le problème posé. Il est essentiel que les données fournies aux machines soient de qualité pour commencer. C'est primordial, car la qualité des données qui entrent dans la machine aura une incidence directe sur la précision du modèle. A cet effet,

Les données doivent : être appropriées, comporter moins de valeurs manquantes possibles, pas de doublons, et couvrir un bon éventail des différentes sous-catégories/classes.

Après avoir recueilli les informations nécessaires, il faut les traiter. Cela peut être fait de la manière suivante : Supprimer toutes les données indésirables, les valeurs vides ou

répétées, convertir les types de données, etc. Il peut être essentiel de réorganiser l'ensemble de données et de modifier les lignes et les colonnes.

Visualiser les données pour comprendre leur structure et le lien entre les différentes variables et classes présentes.

Diviser les données épurées en deux ensembles : un ensemble d'entraînement à partir duquel le modèle apprend, et un ensemble de tests pour vérifier la précision du modèle.

La configuration du modèle

Après avoir appliqué un algorithme d'apprentissage automatique aux données compilées, il est essentiel de choisir un modèle approprié qui est lié à la tâche. En outre, il est essentiel de déterminer si le modèle est conçu pour des données numériques ou catégorielles et de faire la sélection appropriée. La configuration du modèle comprend la sélection des bonnes méthodes d'apprentissage et des hyper paramètres appropriés. Les hyperparamètres sont des variables qui contrôlent le comportement du modèle et qui doivent être ajustées pour obtenir les meilleurs résultats.

L'entraînement du modèle

Une fois que le modèle est configuré, il faudra commencer à l'entraîner. Cela implique l'utilisation des données pour configurer les paramètres du modèle. Les données préparées et transmises au modèle vont lui permettre de faire des prédictions. Le modèle reçoit des données et, au cours d'une période donnée, il est entraîné pour augmenter progressivement sa capacité à répondre à une situation particulière, à résoudre une difficulté complexe. Il est suggéré d'utiliser des données d'entraînement « training set » pour cette étape. La compilation complète des informations collectées est souvent trop lourde et consommatrice de ressources : il est alors adéquat de choisir une fraction de l'ensemble de données pour entraîner le modèle plus efficacement et améliorer ses prévisions. Par ailleurs, l'on peut aussi utiliser des techniques d'optimisation pour trouver les meilleurs paramètres et ajuster le modèle pour obtenir des résultats optimaux.

L'évaluation des performancesL'évaluation des performances

Une fois que le modèle est entraîné, il faudra évaluer sa performance. L'évaluation d'un modèle de Machine Learning consiste à jauger ses métriques, sa matrice de confusion, ses indicateurs de performance clé et la qualité du modèle, puis à vérifier s'il peut atteindre les objectifs commerciaux fixés. Pour cela, il faudra :

l'évaluer avec un ensemble de données et une technique de validation ;

calculer la matrice de confusion dans le cas d'une tâche de classification ; utiliser des techniques de validation croisée si l'approche k-fold est adoptée ;

modifier les hyperparamètres pour maximiser ses performances ;

et comparer le modèle avec un modèle de référence. L'on pourra faire recours à des métriques telles que le rappel et la précision pour mesurer la performance du modèle. Enfin, il faudra s'assurer que le modèle n'est pas surajusté (overfitting) et qu'il fonctionne bien avec des données réelles.

Le test et l'ajustement du modèle. Le test et l'ajustement du modèle.

La dernière étape de l'apprentissage automatique consiste à examiner le modèle dans des conditions réelles. Au cours de cette phase d'évaluation, ce sont essentiellement des données tests qui sont utilisées. Cela permet d'affiner le modèle grâce aux situations ou aux données que l'ordinateur n'a pas encore rencontrées lors de la phase d'apprentissage. De cette manière, l'on pourra évaluer l'efficacité et la performance du modèle dans le contexte de l'entreprise.

4.6 Annexe 2 : Résultats selon les autres métriques de performance

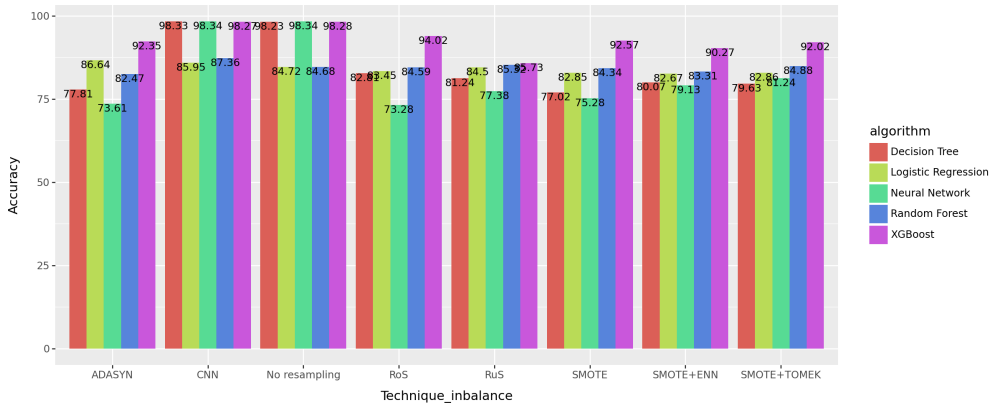

FIGURE 4.6 – Comparaison des modèle selon l'accuracy

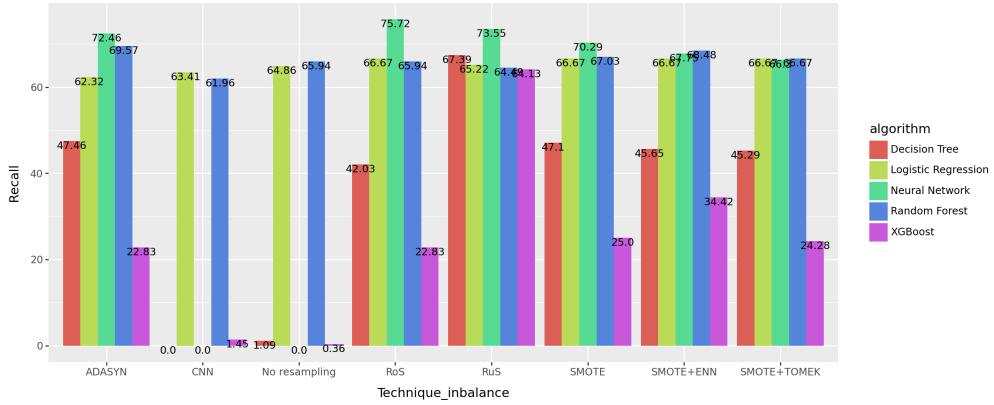

FIGURE 4.7 – Comparaison des modèles selon le recall

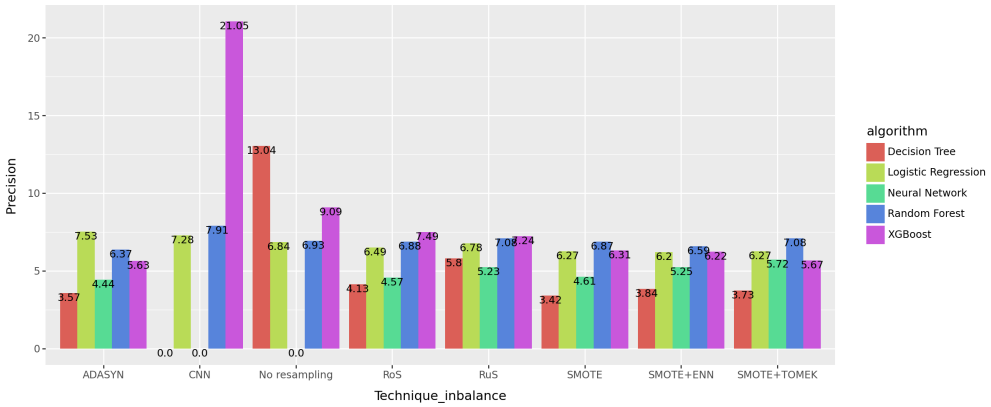

FIGURE 4.8 – Comparaison des modèles selon la précision

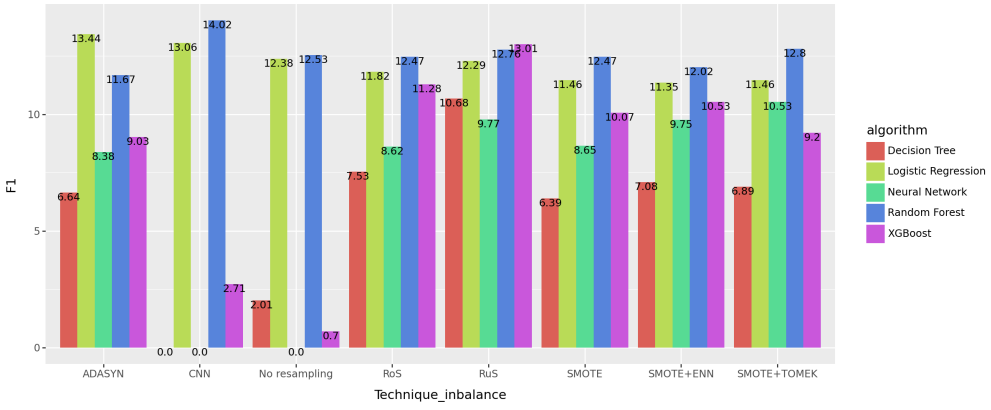

FIGURE 4.9 – Comparaison des modèles selon le F1-score