Rapport sur le Modèle de Scoring

Assoil Ayad

Introduction

La société financière souhaite développer un modèle de scoring pour prédire la probabilité de défaut de paiement des clients demandant des crédits à la consommation. Parallèlement, elle souhaite créer un tableau de bord interactif pour expliquer les décisions d'octroi de crédit et offrir aux clients un accès transparent à leurs informations.

Missions :
- Modèle de scoring : Construire un modèle de scoring prédictif en utilisant différentes sources de données, en s'appuyant sur un kernel Kaggle adapté.

- Dashboard interactif : Créer un tableau de bord interactif pour les chargés de relation client, permettant d'interpréter les prédictions du modèle et d'améliorer la connaissance client, tout en offrant aux clients un accès transparent à leurs informations personnelles.

En réalisant ces deux objectifs, la société pourra évaluer les risques de défaut de paiement et répondre aux attentes de transparence des clients.

1 Méthodologie d'entraînement du modèle

1.1 Préparation des données

Le jeu de données se compose de plusieurs fichiers CSV, le fichier principal d'entraînement comprenant 307,511 enregistrements de clients et 122 variables. La variable cible, nommée `TARGET`, indique la présence ou l'absence de défaut de paiement. Pour enrichir le jeu de données et améliorer la capacité prédictive du modèle, divers ratios ont été introduits, tels que le ratio des jours d'emploi par rapport aux jours de naissance et le ratio entre le revenu total et le montant du crédit. Les étapes de prétraitement des données ont été essentielles. Elles ont inclus l'imputation des valeurs manquantes, l'encodage des variables catégorielles, et la standardisation des variables numériques.

1.2 Exploration des données

Une exploration approfondie des données a été entreprise afin de comprendre les tendances, les distributions, et d'identifier les éventuelles anomalies ou valeurs aberrantes qui pourraient affecter les performances du modèle. Des visualisations et des analyses statistiques ont été utilisées pour cette exploration.

1.3 Modélisation

Plusieurs modèles ont été explorés pour la tâche de prédiction, incluant la Régression Dummy (comme base de référence), la Régression Logistique, RandomForestClassifier, GradientBoosting Classifier, et XGBCClassifier. Le XGBCClassifier, ou XGBoost, s'est distingué des autres grâce à ses performances supérieures. XGBoost est une méthode d'optimisation d'arbres de décision basée sur le gradient, reconnue pour sa vitesse et ses performances, en particulier avec des jeux de données volumineux ayant une structure complexe.

Le déséquilibre des classes dans la variable cible était un défi majeur. Différentes techniques de rééquilibrage ont été testées pour traiter cet enjeu. Chacune de ces techniques a été évaluée avec le XGBCClassifier pour déterminer laquelle améliorait le plus les performances du modèle.

Enfin, afin de garantir que le modèle XGBCClassifier était optimisé pour notre ensemble de données, un `GridSearchCV` a été employé. Cette étape a permis d'ajuster les hyperparamètres du modèle pour atteindre les meilleures performances possibles.

2 Traitement du déséquilibre des classes

Le déséquilibre des classes est un problème courant en apprentissage supervisé où une classe possède un nombre nettement plus faible d'échantillons que l'autre. Dans le contexte de la prédiction de défaut de paiement, cela se traduit par une prédominance d'échantillons de clients qui remboursent leur prêt par rapport à ceux qui font défaut. Une telle distribution peut entraîner des modèles biaisés qui prévoient inexactement la classe minoritaire, en l'occurrence les clients à risque.

Class-Weight : L'utilisation de la pondération des classes est une approche où l'on attribue un poids plus élevé à la classe sous-représentée, rendant le modèle plus sensible aux erreurs sur cette classe. Suite à cette méthode, le modèle a obtenu une AUC de 0.747, avec une précision de 0.173 et un rappel (recall) de 0.496.

Under-sampling : Consiste à réduire le nombre d'échantillons de la classe majoritaire. Bien que cette méthode risque de perdre des informations, elle a démontré une efficacité notable pour le modèle avec une AUC de 0.752, une précision de 0.154, et un rappel impressionnant de 0.592. Cette performance élevée en rappel est cruciale car elle minimise le nombre de faux négatifs, ce qui était l'objectif principal.

Over-Sampling : Vise à augmenter le nombre d'échantillons de la classe minoritaire. Avec cette technique, le modèle a atteint une AUC de 0.776 et une précision parfaite de 1.000, bien que le rappel soit de 0.278.

Lors de l'évaluation du modèle, un faux négatif (FN) a été jugé dix fois plus coûteux qu'un faux positif (FP). Ceci justifie l'importance accordée au rappel lors de l'évaluation des différentes méthodes.

Suite à ces évaluations, le choix s'est porté sur le modèle XGBClassifier avec l'approche d'under-sampling, et des paramètres spécifiques : {'learning_rate' : 0.05, 'max_depth' : 5, 'n_estimators' : 100}. Cette décision a été prise en tenant compte de la performance du modèle en termes de rappel, et de l'importance d'identifier le plus grand nombre possible de clients à risque pour minimiser le coût associé aux faux négatifs.

Fonction coût métier, Algorithme d'optimisation et Métrique d'évaluation

Lors de l'évaluation du modèle, un faux négatif (FN) a été jugé dix fois plus coûteux qu'un faux positif (FP). En effet, lorsqu'une banque évalue la solvabilité d'un client pour l'octroi d'un crédit, elle utilise généralement des modèles prédictifs pour déterminer le risque associé à chaque prêt. Le résultat est souvent une décision binaire : accorder le prêt ou le refuser.

- Faux Négatif (FN) : Un FN se produit lorsqu'un client est classé comme non risqué (c'est-à-dire qu'il remboursera le prêt) par le modèle, alors qu'en réalité, il ne remboursera pas. Les conséquences d'un FN sont généralement plus graves que celles d'un faux positif. En effet, si la banque accorde un prêt à un client en se basant sur une prédiction erronée, elle risque de perdre la totalité du prêt, d'engager des coûts de recouvrement, sans parler des conséquences sur sa réputation et sa solvabilité.

- Faux Positif (FP) : Un FP se produit lorsqu'un client est classé comme risqué par le modèle, alors qu'en réalité, il aurait remboursé le prêt. Bien que cela entraîne un manque à gagner pour la banque en termes d'intérêts non perçus et une éventuelle détérioration de la relation client, le coût d'opportunité est généralement bien inférieur au coût direct d'un prêt non remboursé.

Cependant, il est crucial de comprendre qu'il existe un équilibre à trouver. Si une banque est trop prudente et évite systématiquement les FN, elle pourrait refuser un trop grand nombre de prêts, perdant ainsi des opportunités lucratives et nuisant à sa relation client. C'est ici qu'intervient le concept de seuil.

Le seuil est la valeur limite à partir de laquelle un client est classé comme "risqué". Par défaut, ce seuil est souvent fixé à 0,5 pour les modèles prédictifs binaires. Cependant, en ajustant ce seuil, la banque peut moduler sa tolérance au risque. En diminuant le seuil, la banque acceptera davantage de prêts, mais augmentera le risque de FN. En augmentant le seuil, elle sera plus prudente, mais augmentera le risque de FP. L'ajustement de ce seuil doit donc être réalisé en fonction de la stratégie globale de la banque, de son appétit pour le risque et de ses objectifs financiers.

Tableau de synthèse des résultats

Modèle final choisi : XGBClassifier avec paramètres : {'learning_rate' : 0.05, 'max_depth' : 5, 'n_estimators' : 100} et undersampling.

	AUC	Precision	Recall	F1-score	Custom Score
class_weight	0.747	0.173	0.496	0.256	-37126
Over_sampling	0.776	1.000	0.278	0.435	-36150
Under_sampling	0.752	0.154	0.592	0.244	-36728

Trois méthodes différentes de gestion du déséquilibre des classes sur un ensemble de données : l'utilisation de poids de classes (class weight), le suréchantillonnage (Oversampling) et le sous-échantillonnage (Under sampling).

AUC (Area Under the Curve) : L'AUC, qui est une métrique clé pour les problèmes de classification, est le plus élevé pour la méthode de suréchantillonnage, ce qui indique que cette méthode a le meilleur compromis entre le taux de faux positifs et le taux de vrais positifs. Cependant, les différences entre les trois méthodes ne sont pas très importantes, avec seulement 3% entre la meilleure et la moins bonne.

Precision : La précision est la plus élevée pour le suréchantillonnage, qui atteint une précision parfaite de 1,00. Cela signifie que toutes les prédictions positives du modèle sont effectivement positives. Toutefois, cette valeur parfaite doit être interprétée avec prudence, car elle peut également indiquer que le modèle prédit très peu de cas positifs, comme le suggère le rappel associé.

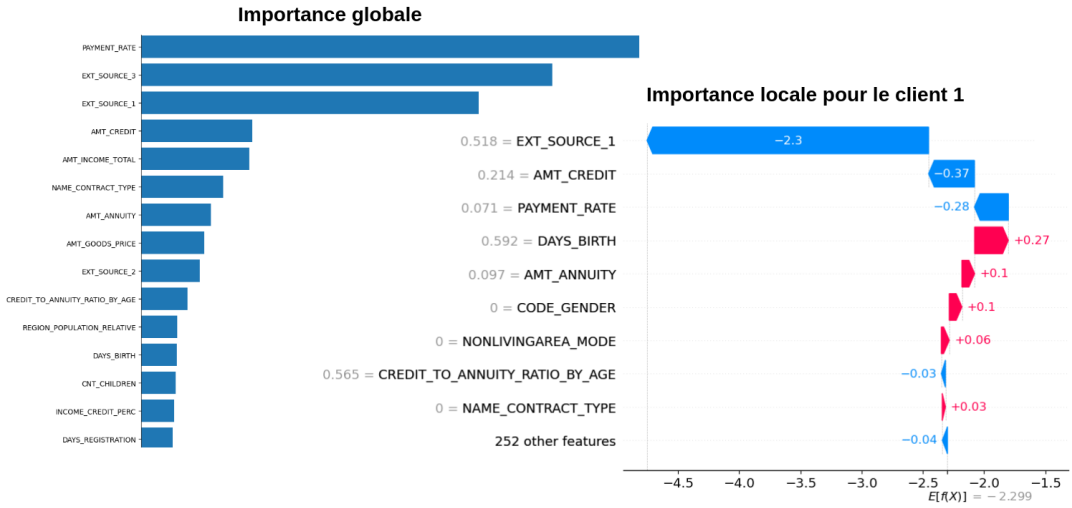
Recall : Le rappel est le plus élevé pour le sous-échantillonnage, ce qui suggère que cette méthode est la meilleure pour identifier la classe minoritaire. C'est important si le coût des faux négatifs est élevé.

F1-score : Le suréchantillonnage a le score F1 le plus élevé, ce qui signifie qu'il a le meilleur équilibre entre la précision et le rappel parmi les trois méthodes. Cependant, la valeur reste relativement faible, suggérant qu'il y a encore des améliorations à apporter.

Custom Score : Le score personnalisé, qui pénalise davantage les faux négatifs, est le plus bas (ce qui est mieux compte tenu de la manière dont il est conçu) pour le suréchantillonnage, bien que les différences entre les méthodes ne soient pas très prononcées.

En conclusion, bien que le suréchantillonnage semble offrir la meilleure performance globale en termes d'AUC et de F1-score, le sous-échantillonnage offre un rappel nettement plus élevé.

Interprétabilité globale et locale du modèle



2.1 Interprétabilité globale :

L'interprétabilité globale se réfère à la capacité de comprendre le modèle dans son ensemble. Cela signifie identifier les caractéristiques ou les variables qui sont les plus importantes pour les prédictions du modèle sur l'ensemble du jeu de données.

Dans un modèle de scoring bancaire basé sur un ensemble de variables, l'interprétabilité globale répondrait à la question : "Quelles sont les variables les plus influentes pour déterminer le score d'un client ?". Les variables PAYMENT_RATE , EXT_SOURCE_3 , EXT_SOURCE_1 et AMT_CREDIT semble globalement etre les plus déterminants dans le scoring.

2.2 Interprétabilité locale

L'interprétabilité locale se concentre sur la compréhension des prédictions pour des observations individuelles. Plutôt que de se demander quelles sont les variables les plus importantes pour l'ensemble du jeu de données, on cherche à comprendre pourquoi un modèle a fait une certaine prédiction pour une entrée spécifique.

Exemple : pour "Individu 1" de notre ensemble de données. son score est négativement influencé par EXT_SOURCE_1 et AMT_CREDIT et PAYMENT_RATE . L'interprétabilité locale répondrait à la question : "Pourquoi le modèle a-t-il donné ce score particulier à l'Individu 1 ?".

Analyse du Data Drift

Dataset Drift

Dataset Drift is NOT detected. Dataset drift detection threshold is 0.5

121	9	0.0744
Columns	Drifted Columns	Share of Drifted Columns

Data Drift Summary

Drift is detected for 7.438% of columns (9 out of 121).

Search

Column	Type	Reference Distribution	Current Distribution	Data Drift	Stat Test	Drift Score
AMT_REQ_CREDIT_BUREAU_QRT	num			Detected	Wasserstein distance (normed)	0.359052
AMT_REQ_CREDIT_BUREAU_MON	num			Detected	Wasserstein distance (normed)	0.281765
AMT_GOODS_PRICE	num			Detected	Wasserstein distance (normed)	0.210785
AMT_CREDIT	num			Detected	Wasserstein distance (normed)	0.207334
AMT_ANNUITY	num			Detected	Wasserstein distance (normed)	0.161102
AMT_REQ_CREDIT_BUREAU_WEEK	num			Detected	Wasserstein distance (normed)	0.15426
NAME_CONTRACT_TYPE	cat			Detected	Jensen-Shannon distance	0.14755
DAYS_LAST_PHONE_CHANGE	num			Detected	Wasserstein distance (normed)	0.138977
FLAG_EMAIL	num			Detected	Jensen-Shannon distance	0.122121
FLAG_DOCUMENT_3	num			Not Detected	Jensen-Shannon distance	0.062496

10 rows 1-10 of 121

L'analyse du data drift du modèle de scoring révèle des informations intéressantes sur la stabilité des données au fil du temps. Le tableau présente une détection de dérive pour 7.438% des colonnes, soit 9 colonnes sur un total de 121. Bien que la détection globale du dataset drift indique qu'il n'est pas détecté (le seuil étant fixé à 0.5), il est essentiel de prendre en compte les colonnes individuelles qui présentent une dérive significative pour assurer la robustesse et la fiabilité du modèle de scoring.

Parmi les colonnes ayant subi une dérive, on remarque que la distance de Wasserstein, une métrique couramment utilisée pour mesurer la différence entre deux distributions de probabilité, est prédominante pour la plupart des variables numériques. Par exemple, les colonnes 'AMT_REQ_CREDIT_BUREAU_QRT' et 'AMT_REQ_CREDIT_BUREAU_MON' et 'AMT_GOODS_PRICE' montrent des dérives importantes, ce qui pourrait suggérer des changements dans les comportements de demande de crédit ou dans la structure des prix des biens au fil du temps.
Une exception notable est la colonne 'NAME_CONTRACT_TYPE', une variable ca-

tégorielle, où la distance de Jensen-Shannon est utilisée pour déterminer la dérive. La présence de dérive dans cette colonne pourrait indiquer une variation dans les types de contrats de crédit privilégiés par les emprunteurs. Finalement, bien que ‘FLAG_DOCUMENT_3‘ ne montre pas de dérive significative, il est essentiel de surveiller de près toutes les variables pour détecter des tendances émergentes qui pourraient affecter la performance du modèle de scoring à l’avenir.

Limites et améliorations possibles

Lors de la réalisation de ce projet de scoring, plusieurs thématiques ont été abordées afin d'assurer une mise en œuvre efficace et sécurisée. Tout d'abord, j'ai exploré l'environnement Mlflow, un système essentiel pour la gestion du cycle de vie des modèles de machine learning. En parallèle, une prise en main d'AWS (Amazon Web Services) a été effectuée, ce qui a permis de mieux appréhender les enjeux et les potentialités de cette plateforme Cloud. Par ailleurs, il a été essentiel de prendre conscience des règles de sécurité afin de garantir la protection des données et des modèles.

Néanmoins, le projet n'a pas été exempt de difficultés. La première barrière a été l'interprétation des variables internes (comme EXT_SOURCE_1) De plus, la modélisation a dû être adaptée en fonction du temps de calcul, limitant ainsi certaines approches potentiellement plus complexes.

Le choix du prestataire Cloud ne s'est pas avéré évident compte tenu des multiples offres sur le marché, et la configuration de l'environnement de travail sur cette plateforme a été plus complexe que prévu. Enfin, il est à noter que les frais associés à certaines ressources et services utilisés ont été plus élevés que prévus.