CY Tech

Master II Mathématiques Appliquées

Année universitaire 2025-2026

Machine Learning Advanced

Performance, robustesse et interprétabilité d'un modèle supervisé

Réalisé par :

Bacarie TCHABO
Zakaria ETTOUHAMI

Encadré par :

M. Lorenzo Aloe

Table des matières

1 Introduction

Ce projet a pour but de concevoir, optimiser et évaluer la performance d'un modèle d'apprentissage supervisé complet. Au-delà de la simple performance prédictive, l'accent est mis sur la fiabilité du modèle : estimation des incertitudes, interprétabilité des décisions (via SHAP) et capacité à détecter et mitiger le *data drift* au cours du temps.

Pour cette étude, nous avons sélectionné le dataset " Credit Card Approval Prediction " disponible sur Kaggle . Ce jeu de données réel simule un problème de gestion du risque de crédit bancaire. l'enjeu ici est d'éffectuer une classification binaire. Il s'agit de prédire si un demandeur de carte de crédit représente un "bon" ou un "mauvais" client en fonction de son profil socio-économique et de son historique de remboursement.

2 Phase 0 — Préparation des données et EDA

2.1 Q0.1 Décrire la cible et le déséquilibre éventuel. Choix des métriques appropriées et justification.

La variable cible (ici, nommée Target) est une variable binaire construite à partir de l'historique des paiements des clients (cette information est issue du fichier credit_record.csv, plus précisement de la colonne nommée STATUS). Nous avons défini le risque de crédit selon une approche "Vintage Analysis" :

- Les mauvais clients son ceux dont la variable Target = 0 : Tout client ayant eu au moins un retard de paiement supérieur à 60 jours (STATUS ≥ 2) durant la fenêtre d'observation.
- Les bon clients son ceux dont la variable Target = 1 : Tout client n'ayant jamais dépassé ce seuil de retard critique.

Quant au déséquilibre de la variable cible, notre dataset présente structurellement un fort déséquilibre. La grande majorité des clients remboursent correctement leurs crédits (Classe 1 majoritaire). Les "Mauvais clients" (Classe 0) représentent la classe minoritaire.

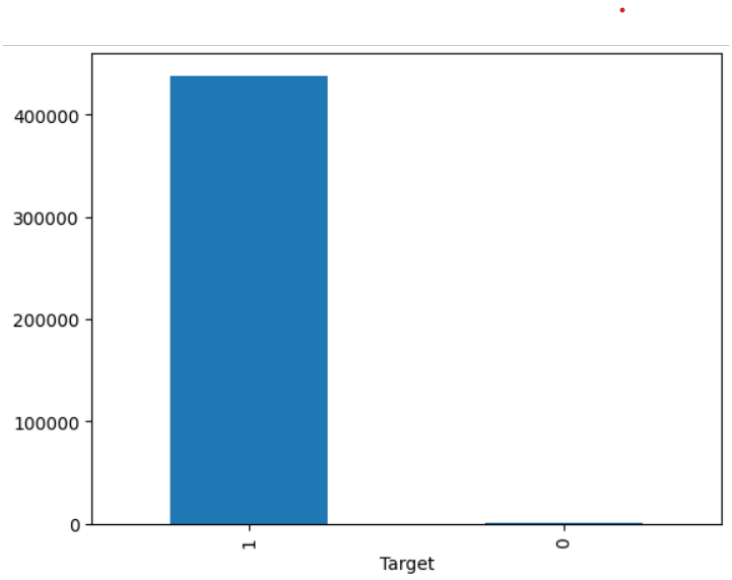

FIGURE 1 – Distribution de la variable cible

2.2 Q0.2 Lister les transformations appliquées. Pourquoi ces choix, impact attendu sur le biais et la variance.

Dans cette étude, nous exploitons deux dataset : un premier dataset contenant différentes informations à propos des clients et un second dataset contenant les informations sur les historiques de remboursement des crédits souscrits. De ce fait, nous avons procédé à plusieurs transformations :

- Nous commençons par procéder à des agrégations temporelles : On souhaite transformer la variable STATUS, qui est en quelque sorte une série temporelle, par des indicateurs statiques (*max*, *count*, etc...). Ces transformations augmentent le biais

car elles impliquent une perte d'information sur la séquentialité des retards. Néanmoins, cela va réduire la variance puisque l'on lisse les comportements erratiques mensuels.

2.3 Q0.3 Montrer corrélations et distributions clés. Quelles hypothèses de modélisation en découlent.

MATRICE DE CORRELATION, etc...

3 Phase 1 — Courbes d'apprentissage

3.1 Q1.1 Interpréter les écarts train vs validation. Sous-apprentissage ou sur-apprentissage.

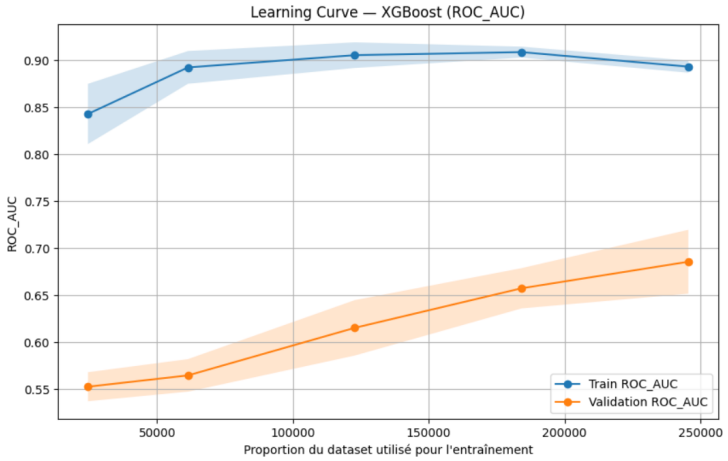

FIGURE 2 – Courbes d'aprentissage

L'analyse de la courbe d'apprentissage du modèle XGBoost révèle deux comportements distincts :

- D'une part, le score d'entraînement reste élevé et stable, oscillant autour de 0.90. Cela indique que le modèle a une capacité suffisante pour capturer la complexité des données (faible biais). Cependant, cela illustre un comportement de sur-apprentissage du modèle.
- D'autre part, le score de validation part d'un niveau bas ($\tilde{0}.55$) mais montre une progression constante pour atteindre environ 0.69 - 0.72.

L'écart important entre le score d'entraînement et le score de validation (environ 0.20 de différence à la fin) est caractéristique d'une variance élevée. Le modèle "sur-apprend" : il mémorise certaines spécificités du jeu d'entraînement qui ne se généralisent pas parfaitement. Cependant, nous ne sommes pas en situation de sous-apprentissage (le score de train est correct). Le modèle est donc assez complexe, mais manque encore de généralisation.

3.2 Q1.2 Quelle quantité de données supplémentaires serait la plus utile. Argumenter via la pente des courbes.

En observant la courbe de validation sur la partie droite du graphique (entre 150 000 et 250 000 échantillons), nous constatons que la pente est toujours strictement positive et elle ne s'aplatit pas. Cela signifie que le modèle continue d'apprendre et de généraliser mieux à mesure qu'on lui fournit plus d'exemples. De plus, la convergence entre les courbes d'entraînement et de validation n'est pas encore atteinte. L'écart se réduit (passant de 0.41 au début à 0.23 à la fin), suggérant que nous n'avons pas atteint la limite asymptotique de performance du modèle. Ainsi, en augmentant le volume de données, nous pourrions augmenter la performance de notre modèle .

4 Phase 2 — Validation croisée et Bootstrap

4.1 Q2.1 Comparer moyenne et écart-type des scores CV vs bootstrap. Les conclusions de performance sont-elles stables

Méthode	Moyenne	Écart-type	Intervalle de confiance approx. ($\pm 2\sigma$)
Stratified cross-validation	0.697	0.044	[0.609 - 0.785]
Bootstrap	0.667	0.017	[0.633 - 0.701]

D'un point de vue performance, on observe un écart d'environ 3 points entre la moyenne de la validation croisée et celle du Bootstrap. Le Bootstrap, en échantillonnant avec remise, laisse souvent de côté des instances difficiles mais s'entraîne sur des doublons, ce qui peut pénaliser un modèle qui a besoin de diversité comme le nôtre.

D'un point de vue variabilité, l'écart-type de la Validation Croisée (0.044) est 2.5 fois supérieur à celui du Bootstrap (0.017).

Au global, les conclusions de performance ne sont pas parfaitement stables. Si le modèle est capable d'atteindre 0.75 sur certains folds, il peut chuter à 0.58 sur d'autres. L'estimateur Bootstrap fournit une vision plus conservatrice et concentrée de la performance réelle.

4.2 Q2.2 Discuter la dépendance des scores aux splits. Montrer une distribution des scores.

L'analyse détaillée des scores par fold en validation croisée révèle une forte sensibilité au découpage des données.

On obtient un score minimal de $\tilde{0}.58$ et un score maximal de $\tilde{0}.77$. Cette grande disparité indique que le jeu de données est hétérogène. Certains sous-ensembles de clients contiennent des motifs de défaut de paiement faciles à détecter, tandis que d'autres contiennent des cas dits "limites" ou bruités où le modèle échoue à généraliser.

5 Phase 3 — Optimisation d'hyperparamètres

5.1 Q3.1 Définir l'espace de recherche et les contraintes. Pourquoi ces bornes.

Nous avons utilisé une recherche aléatoire (via RandomizedSearchCV) pour optimiser le modèle XGBoost. L'espace de recherche a été conçu spécifiquement pour combattre le sur-apprentissage identifié en phase 1 :

- En ce qui concerne la complexité de l'arbre, on fixe le paramètre max_depth etre 2 et 6. Nous avons volontairement contraint la profondeur maximale à des valeurs faibles. Des arbres trop profonds mémorisent le bruit des données, or notre courbe d'apprentissage montrait déjà un écart train/val important.
- D'un point de vue régularisation, on fixe le paramètre reg_lambda entre 0.5 à 4.5 et le paramètre reg _alpha entre 0 et 3.5.L'ajout de pénalités L1 (alpha) et L2 (lambda) force le modèle à sélectionner les caractéristiques les plus fortes et réduit la variance.
- D'un point de vue robustesse aux feuilles, on fixe le paramètre min_child_weight entre 1 et 10. Augmenter ce paramètre empêche le modèle de créer des feuilles spécifiques à un trop petit nombre de clients (outliers).
- Enfin, à propos du nombre d'arbres, on fixe n_estimators entre 100 et 500. On considère que c'est une plage suffisante pour permettre la convergence sans exploser le temps de calcul.

5.2 Q3.2 Produire des validation curves pour 2 hyperparamètres majeurs. Interpréter les zones stables et instables

Nous avons isolé l'impact de la profondeur (le paramètre max_depth) et de la régularisation L2 (le paramètre reg_lambda) sur la performance ROC-AUC.

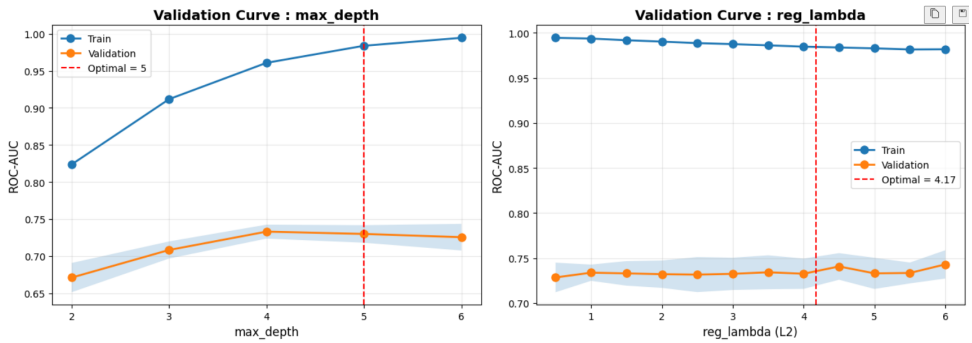

FIGURE 3 – Courbes de validation ROC_AUC en fonction des paramètres max_depth et reg_lambda

En ce qui concerne l'hyperparamètre max_depth, on remarque que la courbe de validation se distingue en trois zones :

- La première zone (les valeurs de Depth < 3) est une zone de sous-apprentissage. Le score est faible, le modèle est trop simple.
- La deuxième zone(les valeurs de Depth entre 3-4) est une zone optimale. On constate que le score de validation atteint son pic (0.735) à une profondeur de 4.
- La troisième et dernière zone (les valeurs de Depth > 4) est une zone d'instabilité/sur-apprentissage. En effet, à partir de la profondeur 5, le score d'entraînement continue de croitre, tandis que le score de validation stagne, voire diminue légèrement. C'est symptomatique d'un sur-apprentissage. Ainsi, cela confirme qu'il faut limiter la profondeur pour généraliser.

En ce qui concerne l'hyperparamètre reg_lambda :

- Tout d'abord, on constate que la courbe est beaucoup plus plate, indiquant que le modèle est relativement stable face aux variations de régularisation
- Ensuite, on observe une légère amélioration du score de validation lorsque Lambda augmente (avec un pic autour de 4.17). Cela valide notre hypothèse : contraindre les poids du modèle aide légèrement à la généralisation sur ce dataset bruité.

5.3 Q3.3 Montrer la meilleure configuration et un intervalle de confiance du score via réévaluation sur multiples splits.

IMPRIMER MEILLEURE CONFIGURATION (best_params) + IDC

6 Phase 4 — Interprétabilité

6.1 Q4.1 Les variables importantes selon permutation coïncident-elles avec SHAP. Analyser divergences

Les features les plus importantes selon la permutation importance sont l'âge du client, la stabilité professionnelle et les variables de revenu. À elles seules, elles expliquent la majeure partie du pouvoir prédictif du modèle. Les variables démographiques familiales ont un impact secondaire mais non négligeable, tandis que les indicateurs simples comme le genre, le nombre d'enfants ou la possession d'un bien immobilier (absurde à première vu mais totalement logique car posséder un bien n'est pas un bon indicateur direct de solvabilité + elle ne dit rien sur la valeur du bien ni si c'est hypothéqué etc...) ont une contribution beaucoup plus faible.

Les valeurs SHAP globales confirment que les variables les plus déterminantes pour le modèle sont :

— *DAYS_BIRTH* (0.538) : l'âge influence fortement la probabilité prédite. Les individus plus jeunes ou appartenant à certaines tranches d'âge apparaissent plus risqués.
— *DAYS_EMPLOYED* (0.472) : la stabilité professionnelle joue un rôle majeur dans la prédiction du risque.
— *AMT_INCOME_TOTAL* : (0.355) : le revenu total est un facteur discriminant important.
— Les autres variables ont un impact plus faible mais demeurent non négligeables.

Ainsi, les variables importantes selon permutation coïncident avec les résultats obtenues via SHAP.

6.2 Q4.2 Illustrer au moins deux dépendances partielles SHAP. Commenter les non-linéarités et interactions.

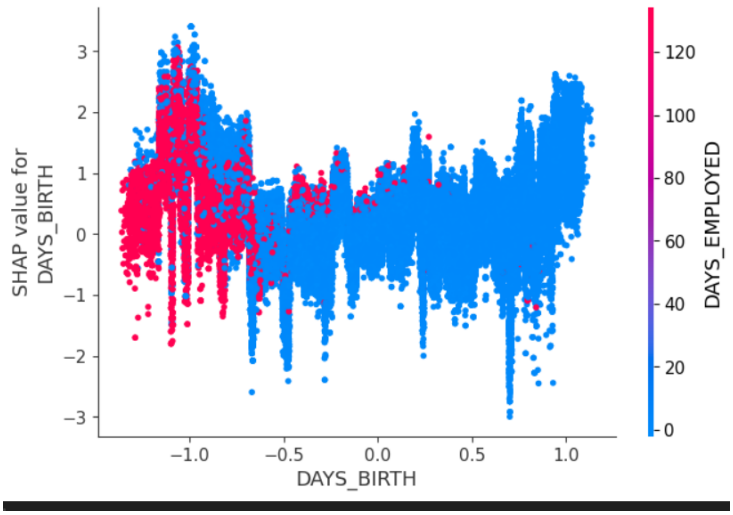

FIGURE 4 – Valeurs SHAP pour la variable *DAYS_BIRTH*

Nous proposons ci-après une analyse de la variable d'age (DAYS_BIRTH) et son interaction avec la variable d'emploi :

Dans un premier temps, on remarque que la relation n'est pas une simple ligne droite. On observe une forte volatilité. Sur la partie droite du graphique (valeurs standardisées positives, correspondant probablement aux clients les plus jeunes dans cet encodage), on voit une chute brutale des valeurs SHAP (jusqu'à -3). Cela indique que le "jeune âge" est un facteur de risque majeur pour le modèle.

Ensuite, on observe que les points roses (haute ancienneté dans l'emploi) sont majoritairement situés sur la partie gauche/centrale avec des SHAP values positives. Cela montre une interaction protectrice : même pour un âge donné, avoir une longue ancienneté professionnelle compense le risque et remonte le score vers "bon client".

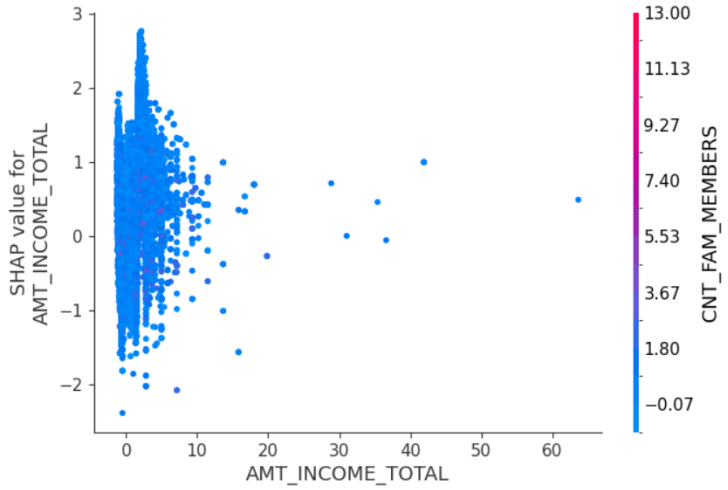

FIGURE 5 – Valeurs SHAP pour la variable *AMT_INCOME_TOTAL*

Nous proposons ci-après une analyse du revenu (la variable AMT_INCOME_TOTAL) :

Contrairement à notre intuition, un revenu très élevé (queue de distribution à droite) ne garantit pas un score infiniment meilleur. Le nuage de points s'aplatit autour de 0 ou légèrement au-dessus. Le modèle a appris que passé un certain seuil, le revenu n'est plus le facteur discriminant principal.

Pour les revenus faibles à moyens (autour de 0 sur l'axe x), la variance verticale est énorme. Cela signifie que pour la majorité de la population, ce n'est pas le revenu qui décide, mais d'autres variables comme la taille de la famille, visible en couleur, ou l'historique de crédit.

6.3 Q4.3 Donner deux explications locales de prédictions. Sont-elles cohérentes avec l'intuition métier.

Nous proposons une première explication locale , on analyse un profil à haut risque - soit un "mauvais client" :

On observe que, un individu situé tout à droite du graphique DAYS_BIRTH (donc, un individu jeune) avec une valeur SHAP de -3.0. Cette prédiction est cohérente avec l'intuition métier. L'instabilité professionnelle combinée à un manque d'historique, du à un jeune âge, est classiquement le profil le plus risqué en banque de détail. Le modèle le sanctionne fortement.

Nous proposons une seconde explication locale B, on analyse un profil sécurisé - autrement dit un "mauvais client" :

Un individu situé à gauche du graphique, autrement dit un individu plus âgé, avec une valeur SHAP élevée de +2.5. Ce point est rose, indiquant une très longue durée d'emploi (correspondant à DAYS_EMPLOYED élevé).

Cette prédiction est cohérente avec l'intuition métier. La stabilité est le premier critère de solvabilité. Le modèle récompense ici la capacité prouvée à conserver un emploi sur le long terme, ce qui corrèle avec des revenus stables pour rembourser.

7 Phase 5 — Simulation et mesure du data drift

7.1 Q5.1 Justifier les variables modifiées, le sens et l'ampleur du drift. Montrer distributions avant et après.

On prends le choix de modifier deux variables :

- On modifie l'age du client : c'est un indicateur de risque important dans le cas des crédit vu que les clients jeunes (20-30 ans) sont plus à même de ne pas rembourser leur crédit. Ils ont généralement des revenus instables et peu d'historique en terme de crédit. Les client âgés eux sont moins à même de ne pas rembourser car ils ont des revenus stable et ont souvent un patrimoine. On modifie la variable pour simuler des phénomènes tel que le vieillissement démographique, et pour caractériser le fait que les jeunes demandent moins de crédits mais aussi que l'accès au crédit est plus difficile pour les jeunes.
- On modifie le revenu des clients : par cela, on souhaite caractériser le fait que les salaires varient parfois avec l'inflation, ce qui implique une augmentation des revenus des clients. On veut aussi caractériser des éventuelles augmentations salariales ou encore une croissance économique.

7.2 Q5.2 Synthétiser un tableau des métriques de drift par variable. Quelles mesures sont les plus sensibles ici et pourquoi.

Le tableau ci-dessous synthétise les métriques de dérive calculées pour les variables ayant subi des modifications.

Variable	KS	Wasserstein
DAYS_BIRTH	0.7543	1.5
AMT_INCOME_TOTAL	0.5659	1.0

On observe que la statistique KS est particulièrement sensible dans ce scénario. Avec un score de 0.7543 pour DAYS_BIRTH, le test indique un écart maximal extrême entre les fonctions de répartition cumulées de l'échantillon de référence et de l'échantillon drifté. Cela traduit un changement radical de la forme de la distribution.

La Distance de Wasserstein, quant à elle, capture l'ampleur du "déplacement" des données. Une valeur de 1.5 pour DAYS_BIRTH confirme que non seulement la forme a changé, mais que la masse des données s'est déplacée significativement dans l'espace des valeurs. C'est une métrique qui complète le KS en quantifiant l'effort nécessaire pour transformer la distribution driftée en distribution d'origine. Ici, ces valeurs très élevées signalent une alerte critique : le modèle opère sur une population qui ne ressemble plus à celle de l'apprentissage.

7.3 Q5.3 Évaluer la dégradation des performances baseline sur le set drifté. Quelles métriques chutent le plus, expliquer.

L'impact du data drift sur la fiabilité du modèle est immédiat et sévère. En évaluant notre modèle (optimisé en Phase 3) sur le jeu de données drifté, nous constatons une chute brutale des capacités prédictives.

Performance avant drift	0.7416
Performance aprés drift	0.6158
Dégradation :	0.1258

Le score ROC AUC passe de 74.2% à 61.6%). Cette dégradation de 12.6 points s'explique par la nature des variables touchées : DAYS_BIRTH et AMT_INCOME_TOTAL sont des prédicteurs clés dans le scoring de crédit, l'âge et le revenu étant corrélés à la solvabilité.

Le modèle a appris des règles de décision spécifiques qui deviennent obsolètes ou trompeuses suite au drift. Le covariate shift sans réentraînement entraîne donc une inadéquation des frontières de décision.

8 Phase 6 — Mitigation du drift

8.1 Q6.1 Définir des critères objectifs de sélection pour la suppression. Impact sur performance et stabilité.

Pour mitiger le drift, la première stratégie envisagée est la suppression pure et simple des variables instables. En supprimant DAYS_BIRTH et AMT_INCOME_TOTAL, nous acceptons de perdre de l'information prédictive au profit de la stabilité. Nous attendons une baisse de la performance théorique maximale, mais une meilleure robustesse dans le temps, car le modèle ne se basera plus sur des données mouvantes.

8.2 Q6.2 Décrire le protocole de réentraînement sans fuite d'information. Montrer les résultats sur validation interne.

Après suppression des variables driftés, on obtient un score moyen de ROC_AUC de 68.73% . Ce score est nettement plus faible que les 73% obtenus après optimisation des hyperparamètres

En appliquant une mitigation, on obtient un score moyen de 68.1%, ce qui est également plus faible que le score obtenue en entraînant le modèle sur le dataset d'origine.

La stratégie de suppression permet de "récupérer" environ la moitié de la performance perdue (+7.15 points par rapport au set drifté). Le modèle est de nouveau utilisable, bien que moins performant que dans l'environnement idéal d'origine.

8.3 Q6.3 Comparer les stratégies. Quel compromis coût, complexité, performance recommander

Face au data drift, trois grandes postures stratégiques s'offrent à nous :

— La première posture serait de conserver le modèle tel quel. Ce qui résulte à une performance médiocre (un score AUC $\tilde{6}1\%$).

— La deuxième posture serait d'effectuer un ré entraînement complet, attendre d'avoir les nouveaux labels pour ré entraîner le modèle sur les nouvelles distributions. Cependant, cette approche est coûteuse et complexe à mettre en œuvre.

— Enfin, la troisième posture serait de supprimer les features instables. C'est une solution immédiate et peu coûteuse.

9 Conclusion

Ce projet de Machine Learning avancé a permis de concevoir une chaîne de modélisation complète pour l'évaluation du risque de crédit, allant de l'analyse exploratoire à la sécurisation du déploiement. L'utilisation de l'algorithme XGBoost, optimisé rigoureusement, a fourni une performance de référence solide avec une ROC AUC proche de 74%, tout en garantissant une transparence décisionnelle grâce aux méthodes d'interprétabilité qui ont validé la cohérence métier des indicateurs socio-économiques. Cependant, l'enseignement majeur de cette étude réside dans la mise en évidence de la vulnérabilité des modèles face au data drift, illustrée par une chute sévère de la performance suite à la simulation de changements démographiques. La stratégie de mitigation par suppression des variables instables a prouvé son efficacité en restaurant la stabilité du système à un niveau de 68.7%, soulignant ainsi l'importance cruciale de privilégier la robustesse opérationnelle sur la performance pure et la nécessité absolue d'un monitoring continu pour garantir la viabilité du modèle à long terme.