ANNEXE 1 : PAGE DE GARDE

MÉMOIRE

Année universitaire 2024 / 2025

NOM et Prénom de l'étudiant : WANE Ibrahima

Formation : M2 MIAGE ID

Entreprise d'accueil : Société Générale

Titre du mémoire : L'explicabilité dans les systèmes de recommandation

Tuteur pédagogique: Véronique Fournel

Tuteur entreprise : Carole Beroldy

Résumé (10 lignes) :

Ł'explicabilité dans les systèmes de recommandation

Ibrahima WANE

Master MIAGE , Informatique Décisionnelle
Université Paris Dauphine , PSL
Société Générale

Tuteurs :
Véronique FOURNEL
Carole BEROLDY

8 septembre 2025

Résumé

Ce mémoire porte sur l'explicabilité des systèmes de recommandation bancaires, avec une étude de cas appliquée au moteur de recommandation **MRI** déployé par la Société Générale. Dans un contexte marqué par des contraintes réglementaires et organisationnelles, il ne suffit pas qu'un algorithme soit performant : il doit également fournir des justifications claires et compréhensibles pour ses décisions.

L'étude explore les limites actuelles du MRI en matière d'interprétabilité, puis propose l'intégration de la méthode **SHAP** (SHapley Additive exPlanations) afin d'attribuer à chaque variable une contribution marginale à une recommandation. Un pipeline expérimental a été conçu et testé, permettant d'obtenir à la fois des explications locales (niveau client-produit) et des explications globales (tendances agrégées).

Les résultats montrent que si SHAP fournit des explications pertinentes, il présente également des limites techniques : coût computationnel élevé, instabilité en présence de variables corrélées, et difficulté d'interprétation en haute dimension. Ces limites sont discutées en profondeur, et des solutions comme la réduction dimensionnelle, l'agrégation de variables ou la combinaison avec d'autres méthodes explicatives (Permutation Importance, LIME) sont envisagées.

Enfin, l'étude met en évidence l'intérêt opérationnel de l'explicabilité globale, notamment via la mesure de l'*effet de levier*, qui relie les variables explicatives à des indicateurs marketing concrets. Cette démarche contribue à renforcer la transparence et la confiance dans les outils d'IA en banque de détail, tout en améliorant leur adoption par les conseillers et leur efficacité dans les campagnes commerciales.

Mots-clés : Explicabilité, SHAP, systèmes de recommandation, intelligence artificielle, banque, appétence produit.

Abstract

This thesis focuses on the explainability of recommendation systems in banking, with a case study applied to the **MRI** recommendation engine deployed by Société Générale. In a context shaped by strong regulatory and organizational constraints, an algorithm's predictive performance is not sufficient : its decisions must also be transparent and understandable.

The study first highlights the current limitations of MRI regarding interpretability, then proposes the integration of the **SHAP** (SHapley Additive exPlanations) method to assign each feature a marginal contribution to recommendations.

An experimental pipeline was designed and tested, producing both local explanations (client-product level) and global explanations (aggregated trends).

Results demonstrate that while SHAP provides relevant insights, it also presents technical limitations : high computational cost, instability with correlated features, and challenges in high-dimensional interpretation.

These issues are thoroughly discussed, with mitigation strategies such as dimensionality reduction, feature grouping, and hybrid approaches with other explainability methods (Permutation Importance, LIME).

Finally, the study emphasizes the operational value of global explainability, notably through the measurement of the *leverage effect*, which connects SHAP results to concrete marketing performance indicators.

This work contributes to enhancing transparency and trust in AI tools in retail banking, while fostering their adoption by advisors and their efficiency in targeted campaigns.

Keywords : Explainability, SHAP, recommender systems, artificial intelligence, banking, product propensity.

Remerciements

Je souhaite exprimer ma profonde gratitude à l'ensemble des personnes qui m'ont accompagné et soutenu tout au long de la réalisation de ce mémoire.

Je tiens tout d'abord à remercier chaleureusement Carole Beroldy, ma tutrice en entreprise, pour son ouverture d'esprit, sa patience et la confiance qu'elle m'a témoignée. Son exigence et sa manière de me challenger sur les missions confiées m'ont permis de progresser et d'apprendre énormément. Mes remerciements vont également à Marianne Bioy, qui était initialement ma tutrice entreprise avant une longue absence, et qui m'a beaucoup aidé pour mon intégration au sein de l'équipe.

Je remercie également Véronique Fournel, ma tutrice pédagogique, pour sa disponibilité, ses conseils avisés et ses remarques toujours pertinentes, qui ont contribué à enrichir et à affiner ce travail.

Un grand merci à toute l'équipe du projet MRI, en particulier Marc-Antoine Iehl, responsable du projet MRI, pour sa disponibilité et les points réguliers que nous avons partagés. Ses orientations et ses conseils m'ont été d'une aide précieuse dans l'avancement de ce mémoire.

Enfin, je ne saurais terminer sans exprimer toute ma reconnaissance à mes parents ainsi qu'à mes petits frères et sœurs. Leur soutien inconditionnel et l'amour qu'ils me portent ont été une source de motivation constante tout au long de ce parcours.

Table des matières

Table des figures

Liste des abréviations

DAM	Data Science & Analytics for Marketing
DIP	Data Intelligence & Products
DRO	Digital, Relation client à distance et Open-banking
LIME	Local Interpretable Model-agnostic Explanations
MRI	Moteur de Recommandation Individus
MRH	Multirisque Habitation
PEL	Plan Épargne Logement
SHAP	SHapley Additive exPlanations
SVM	Support Vector Machine
U2CMS	User and Item-aware Contextualized Multi-hop Sequence

I | Introduction générale

1.1 Contexte professionnel et mission d'apprentissage

Mon apprentissage s'est déroulé au sein de la Société Générale, au cœur de la Direction **DIP (Data Intelligence & Products)**. Cette direction occupe une position stratégique dans l'organisation de la banque, en pilotant les initiatives autour de la valorisation des données à grande échelle. Elle regroupe des compétences pluridisciplinaires en data science, ingénierie, analyse métier et gouvernance de l'information, avec pour ambition de transformer les gisements de données internes en leviers concrets d'aide à la décision, d'automatisation et de personnalisation.

Les missions principales de la DIP s'articulent autour de quatre grands axes. Le premier concerne l'analyse stratégique, c'est-à-dire la capacité à produire des insights exploitables à partir de grands volumes de données, pour mieux piloter les activités de la banque et éclairer les choix des directions métier. Le second axe est celui de la modélisation avancée, qui recouvre le développement d'algorithmes de machine learning et de modèles prédictifs, adaptés aux problématiques spécifiques des différentes lignes métiers (risques, marketing, conformité, relation client, etc.). Vient ensuite le déploiement opérationnel d'outils, notamment des tableaux de bord, reportings dynamiques et visualisations interactives, destinés à rendre les données intelligibles et activables. Enfin, la proximité métier constitue un principe fondamental : l'ensemble des projets sont conçus en lien étroit avec les utilisateurs finaux, afin de garantir leur pertinence fonctionnelle, leur valeur ajoutée réelle, et leur adoption dans les processus existants.

Au sein de cette direction, j'ai intégré le service **DAM (Data Science & Analytics for Marketing)**, une entité experte dans l'exploitation des données au service du marketing relationnel. DAM est chargée de segmenter la clientèle, de modéliser les comportements utilisateurs, d'optimiser le ciblage des campagnes commerciales, et d'an-

ticiper les besoins produits ou services des clients à travers des modèles prédictifs. Elle collabore au quotidien avec les équipes du marketing stratégique, les responsables d'offre, les concepteurs d'outils CRM, et les canaux de distribution (agences, banque en ligne, centres de relation client).

Dans ce contexte, l'un des actifs majeurs de DAM est le moteur de recommandation **MRI (Moteur de Recommandation Individus)**, un système algorithmique conçu pour proposer à chaque client particulier de la Société Générale une sélection personnalisée de produits financiers. Ce moteur repose sur un modèle de scoring avancé, nourri de données comportementales, transactionnelles et historiques, et enrichi par des règles métier. Il constitue aujourd'hui un outil central d'aide à la décision, utilisé par les conseillers dans leurs interactions avec les clients et intégré à certains processus de communication commerciale.

Le fonctionnement du MRI repose sur un principe simple : chaque semaine, il calcule pour l'ensemble de la clientèle active les trois produits les plus pertinents à recommander, en fonction de leur probabilité de souscription. Ces produits peuvent inclure des offres d'épargne, de crédit, d'assurance ou de services bancaires. Les recommandations sont ensuite filtrées par un ensemble de règles d'éligibilité, d'exclusion ou de priorisation métier, puis restituées dans les outils opérationnels sous forme de fiches actionnables. Ce moteur incarne ainsi une application concrète et industrialisée de la science des données, au service de la performance commerciale et de la relation client personnalisée.

Ma mission d'apprentissage s'est inscrite dans le prolongement de cet outil. Elle visait initialement à comprendre le fonctionnement du moteur MRI dans ses différentes composantes, algorithmique, métier, opérationnelle, et à en évaluer la valeur d'usage. En analysant à la fois la logique de scoring, les règles d'éligibilité, le cycle de mise à jour des données et les modalités de restitution dans les outils, j'ai progressivement orienté mon travail vers une problématique précise, mais centrale dans un contexte bancaire : la question de l'explicabilité des recommandations.

En effet, dans le cadre réglementaire et organisationnel actuel, il ne suffit pas qu'un moteur de recommandation soit performant du point de vue de la précision prédictive. Encore faut-il que ses suggestions puissent être justifiées de manière claire, intelligible et fiable. C'est particulièrement vrai dans un environnement de conseil bancaire, où les décisions doivent être comprises, expliquées et assumées par les conseillers, puis acceptées

par les clients. Or, plusieurs signaux, issus de l'analyse documentaire, de l'étude des règles métier, et de premières observations, laissent entrevoir un décalage entre la complexité algorithmique du moteur et la lisibilité de ses résultats en situation réelle.

Ce constat a structuré l'évolution de ma mission autour d'un double objectif. D'une part, il s'est agi d'analyser les limites actuelles de l'explicabilité dans le MRI : nature et couverture des règles explicites ainsi que l'absence d'explication sur les scores issus du modèle. D'autre part, j'ai exploré la possibilité d'introduire une forme d'explicabilité algorithmique post hoc, en m'appuyant sur des techniques issues du champ de l'eXplainable Artificial Intelligence (XAI). Parmi les outils mobilisables, la méthode **SHAP (SHapley Additive exPlanations)**, que nous évoquerons plus loin, a retenu une attention particulière, en raison de sa capacité à attribuer à chaque variable une contribution marginale à une prédiction donnée, ce qui ouvre la voie à des explications localisées, individualisées et potentiellement interprétables.

Le présent mémoire s'inscrit dans cette dynamique. Il vise à articuler rigoureusement une analyse critique du moteur MRI, une étude des limites de son exploitabilité actuelle, et une proposition de solution explicable, intégrable au système existant. Plus largement, il ambitionne de contribuer à la réflexion autour de la conformité, de la transparence et de la responsabilité des systèmes d'intelligence artificielle déployés dans un contexte sensible tel que la banque de détail. Il s'adresse ainsi à la fois aux professionnels des données, aux concepteurs d'outils, et aux métiers qui en sont les utilisateurs quotidiens.

1.2 Présentation synthétique du moteur MRI

Déjà évoqué en introduction comme un dispositif central dans l'arsenal analytique de l'entité DAM, le moteur MRI mérite à présent un examen plus détaillé, tant il structure aujourd'hui une part essentielle des recommandations personnalisées adressées aux clients particuliers de la banque. Au-delà de sa finalité opérationnelle, c'est-à-dire proposer à chaque individu une sélection pertinente de produits bancaires, le MRI incarne une mise en œuvre avancée des capacités de la data science dans un environnement réglementé et à haute exigence métier. Comprendre son fonctionnement permet de saisir à la fois la richesse de ses apports et les limites actuelles de son explicabilité. Cette section propose ainsi d'en décrire l'architecture algorithmique, les logiques de calcul mises en œuvre, son

intégration au sein des outils métiers, ainsi que les mécanismes de filtrage et de restitution qui accompagnent sa mise en production.

Le MRI repose sur un principe de scoring hebdomadaire appliqué à l'ensemble de la clientèle de la banque de détail. Chaque semaine, pour chaque client actif, il évalue la probabilité de souscription à une gamme prédéfinie de produits bancaires, couvrant des catégories aussi diverses que les livrets d'épargne, les crédits à la consommation, les assurances ou encore les solutions de banque au quotidien. À l'issue de ce calcul, il sélectionne les trois produits présentant les scores les plus élevés, et les restitue dans les interfaces des conseillers ou dans les outils de marketing relationnel, avec pour finalité d'aider à la priorisation commerciale.

Sur le plan algorithmique, le cœur du MRI est inspiré du modèle U2CMS (User and Item-aware Contextualized Multi-hop Sequence), décrit initialement par Yang, Jang et Kim (2020) [1]. Dans cet article, les auteurs présentent le modèle conçu pour dépasser certaines limites des approches traditionnelles de recommandation. Les systèmes classiques, qu'ils soient collaboratifs ou basés sur le contenu, peinent en effet à intégrer la dimension temporelle et séquentielle des comportements utilisateurs, alors même que ceux-ci suivent souvent des trajectoires logiques dans le temps. L'apport majeur de l'U2CMS est précisément d'introduire une modélisation hybride, où les interactions passées des utilisateurs et des items sont enrichies par une prise en compte explicite des dépendances séquentielles de plus haut niveau. Le modèle repose ainsi sur une articulation entre un module de similarité inspiré des techniques factorielles (FISM), un module collaboratif adapté aux problèmes de sparsité des données, et un module séquentiel basé sur des chaînes de Markov multi-sauts. Cette combinaison permet d'exploiter simultanément les préférences de long terme, les proximités entre clients et produits, ainsi que la dynamique des comportements dans le temps. Les résultats empiriques rapportés par les auteurs mettent en évidence une amélioration substantielle des performances de recommandation par rapport aux approches existantes, y compris certains modèles neuronaux profonds, tout en offrant une meilleure robustesse face aux données clairsemées.Ce modèle repose sur l'agrégation de plusieurs sources d'informations complémentaires pour modéliser la propension d'un client à souscrire un produit. La logique retenue est hybride, combinant de façon pondérée trois composantes principales : une composante collaborative, une composante projective, et une composante séquentielle.

La première composante, dite collaborative, s'appuie sur l'analyse des similarités entre clients au regard des produits qu'ils détiennent ou ont souscrits récemment. Cette logique

est dérivée des techniques classiques de filtrage collaboratif, mais enrichie pour tenir compte de contraintes métiers spécifiques. Elle permet d'identifier, dans la base client, des profils proches, et d'en inférer des tendances de souscription pertinentes.

La seconde composante consiste en une projection des caractéristiques du client dans un espace réduit, obtenu via une analyse en composantes principales (ACP). L'objectif est ici de capter les dimensions latentes du profil client, en s'affranchissant des corrélations fortes entre variables brutes. Ce sous-espace facilite la généralisation du modèle, en réduisant le bruit et en mettant en avant des dimensions synthétiques pertinentes pour la modélisation de l'appétence.

La dernière composante s'appuie sur une chaîne de Markov d'ordre quatre, modélisant les trajectoires de souscription successives observées dans l'historique client. Ce choix permet de capter les dépendances séquentielles de plus haut niveau entre produits : par exemple, la probabilité qu'un client ayant souscrit à un livret jeune, puis à une carte premium, puis à une assurance habitation, présente un intérêt pour une solution d'épargne retraite. Cette logique multi-hop autorise ainsi une modélisation plus fine des parcours utilisateurs.

Formellement, le score d'appétence d'un client u pour un produit i au temps t, noté $P_{u,i,t}$, s'exprime par la formule suivante :

$$P_{u,i,t} = \sum_{k=1}^{k_{\text{mod}}} w^B x^B_{u,i,k} + w^P \sum_{p=1}^{k_{\text{pca}}} x^P_{u,i,p} + w^{MC} \sum_{c=1}^{k_{\text{mc}}} x^L_{u,i,c}$$

où $x^B_{u,i,k}$ désigne les similarités collaboratives, $x^P_{u,i,p}$ les variables projetées issues de l'ACP, et $x^L_{u,i,c}$ les contributions séquentielles issues de la chaîne de Markov.

Les pondérations w^B, w^P, w^{MC} sont apprises automatiquement par le modèle via une fonction de perte logistique de type `BCEWithLogitsLoss`, adaptée au déséquilibre entre produits populaires et produits rares. Ce modèle est réentraîné régulièrement, sur la base d'un historique mobile de vingt-quatre mois de données, et mis à jour hebdomadairement.

Ce scoring, aussi précis soit-il, ne constitue que la première étape du pipeline MRI. Une phase de filtrage métier est ensuite appliquée afin de garantir la conformité réglementaire, l'éligibilité des clients aux produits proposés, et la cohérence des recommandations au regard des politiques commerciales en cours. Cette phase repose sur un ensemble de règles

déterministes, formalisées et maintenues par les équipes marketing et conformité. Ces règles peuvent concerner l'âge, la situation géographique, le statut client, la présence ou non de certains produits en portefeuille, des événements récents (clôture, incident), ou encore des combinaisons logiques d'exclusion. Loin d'être secondaires, ces règles métiers jouent un rôle fondamental dans la crédibilité du système. Elles assurent que le moteur ne propose pas, par exemple, un prêt immobilier à un mineur, ou une carte haut de gamme à un client en situation financière fragile. Toutefois, comme nous le verrons dans les sections suivantes, elles introduisent également des limites importantes en matière d'explicabilité, dans la mesure où elles sont déconnectées du modèle de scoring lui-même.

Une fois le score calculé et les règles de filtrage appliquées, le MRI sélectionne les trois produits présentant la meilleure combinaison de pertinence et de conformité, et génère une recommandation finale. Celle-ci est intégrée dans différents canaux de restitution. Dans les outils CRM utilisés par les conseillers en agence, une fiche par client présente les produits recommandés, la date de calcul du score, ainsi qu'un éventuel motif explicatif issu des règles métier. Ces fiches sont accessibles avant un rendez-vous, mais aussi en rebond lors d'une interaction spontanée avec le client. Les conseillers disposent également de ces informations dans leurs interfaces, leur permettant d'adapter leur discours commercial. Enfin, certains parcours digitaux (espace client en ligne, notifications push, campagnes d'emailing) exploitent également les résultats du MRI pour personnaliser les messages envoyés aux clients.

Il est important de souligner que le MRI n'est pas un prototype ou un outil en phase expérimentale. Il s'agit d'un moteur pleinement déployé, industrialisé, et intégré dans les processus quotidiens de la banque de détail. Son maintien en condition opérationnelle repose sur une équipe dédiée, assurant le monitoring, la mise à jour des modèles, la gestion des incidents, et l'évolution des règles métier. Cette maturité en production témoigne de la confiance placée dans le dispositif, mais accroît aussi les exigences en matière de transparence, de robustesse et de compréhension par les utilisateurs finaux.

Toutefois, comme toute solution à base d'intelligence artificielle, il soulève des questions d'interprétabilité, de responsabilité et de contrôle. En particulier, l'analyse critique de ses mécanismes d'explication, ou de leur absence, révèle des tensions entre performance technique et intelligibilité métier, que le présent mémoire se propose d'explorer plus avant.

1.3 Problématique identifiée : l'enjeu de l'explicabilité dans les systèmes de recommandation bancaires

Dans l'univers des systèmes de recommandation à vocation industrielle, une tension fondamentale s'installe entre deux dynamiques qui, à première vue, peuvent sembler difficilement conciliables. D'une part, les algorithmes d'apprentissage automatique, devenus incontournables dans la personnalisation des services, cherchent à maximiser la performance prédictive. D'autre part, les contextes d'usage dans lesquels ces systèmes opèrent, notamment dans les environnements réglementés comme la banque, exigent une transparence accrue, une intelligibilité des décisions automatisées, et une capacité à justifier les recommandations formulées. Ce paradoxe entre performance et explicabilité est au cœur de la problématique traitée dans ce mémoire, et se manifeste avec une acuité particulière dans le cas du moteur MRI.

Le moteur MRI, comme nous l'avons vu précédemment, est un outil puissant de ciblage algorithmique, capable d'identifier chaque semaine les trois produits bancaires les plus pertinents à recommander à chaque client particulier, à partir de l'analyse croisée de données historiques, comportementales et transactionnelles. S'il se distingue par sa robustesse technique, sa couverture populationnelle étendue, et son intégration fluide dans les dispositifs opérationnels de la banque de détail, il souffre néanmoins d'une faiblesse importante : l'opacité de ses décisions. En effet, les recommandations générées par le moteur, bien que pertinentes sur le plan statistique, apparaissent souvent comme des « boîtes noires » aux yeux des utilisateurs finaux, en particulier les conseillers chargés de les interpréter, de les reformuler, et de les intégrer dans la relation client.

Ce déficit d'explicabilité ne constitue pas un simple défaut esthétique ou ergonomique. Il touche au cœur de la confiance que l'on peut accorder à une recommandation algorithmique, en condition réelle d'usage. Il est susceptible de générer de la défiance, de l'incompréhension, voire de l'inaction. Dans un cadre aussi sensible que celui du conseil bancaire, où chaque recommandation peut avoir des implications financières, juridiques ou personnelles pour le client, l'absence d'une justification claire, compréhensible et contextuelle constitue un frein majeur à l'appropriation du moteur par les équipes métiers, et plus largement à son efficacité globale.

Ce chapitre a pour ambition de formuler précisément la problématique centrale du

mémoire, à savoir : *comment améliorer l'explicabilité du moteur MRI, dans le respect des contraintes techniques, réglementaires et fonctionnelles qui le structurent* ? Pour ce faire, nous commencerons par clarifier le concept d'explicabilité tel qu'il est défini dans la littérature scientifique sur l'intelligence artificielle. Nous montrerons ensuite pourquoi cette exigence prend une importance particulière dans le secteur bancaire, en lien avec les obligations réglementaires imposées par le RGPD et les orientations à venir de l'AI Act. Enfin, nous analyserons les limites concrètes du moteur MRI sur ce point, afin de poser les bases d'une réponse méthodologique adaptée, qui sera développée dans les chapitres suivants.

L'explicabilité : un enjeu central dans les systèmes intelligents

Le terme *explicabilité* (ou *explainability* en anglais), devenu central dans le champ de l'intelligence artificielle, désigne la capacité d'un système algorithmique à produire, en plus de ses décisions, des éléments de justification permettant à un humain d'en comprendre le fonctionnement ou, à tout le moins, d'en interpréter les sorties. Il s'agit d'un champ de recherche à part entière, regroupé sous l'acronyme *XAI* (*eXplainable Artificial Intelligence*), et qui vise à concilier la complexité croissante des modèles d'IA, en particulier ceux reposant sur l'apprentissage profond ou les architectures séquentielles, avec la nécessité de transparence, de traçabilité et de compréhension par des utilisateurs humains.

Plusieurs auteurs ont tenté de préciser les contours de ce que recouvre l'explicabilité. Lipton (2018) distingue notamment la *transparence*, qui concerne les propriétés internes du modèle (par exemple, la lisibilité d'une régression linéaire ou d'un arbre de décision), et la *post-hoc interpretability*, qui regroupe les méthodes permettant de produire une explication *a posteriori*, sans que le modèle sous-jacent soit nécessairement interprétable par construction. Dans le cas de modèles dits « boîte noire », comme les réseaux de neurones profonds, les modèles séquentiels, ou encore les modèles de scoring complexes, les méthodes d'explicabilité post hoc constituent souvent la seule voie praticable.

De manière plus opérationnelle, Doshi-Velez et Kim (2017) [2] proposent de considérer l'explication comme un artefact communicable entre un système d'IA et un utilisateur humain, visant à répondre à trois questions : *que fait le système* ? *pourquoi l'a-t-il fait* ? et *que pourrait-il faire autrement* ? Dans cette perspective, l'explication n'est pas uniquement une propriété technique, mais un acte de médiation, entre l'intentionnalité supposée d'un

système algorithmique et les attentes cognitives, pratiques ou normatives d'un utilisateur.

Dans les systèmes de recommandation, cette exigence d'explicabilité se heurte à plusieurs défis spécifiques. Contrairement aux systèmes de classification ou de décision binaire, où la sortie du modèle peut être interprétée en termes de seuil, de score ou de probabilité, une recommandation est une suggestion ouverte, souvent non contraignante, et susceptible d'être évaluée de manière subjective. Elle est contextualisée dans un parcours, influencée par les préférences passées, et sujette à l'interprétation du récepteur. Dès lors, l'explication ne peut se limiter à une justification mécanique : elle doit être compréhensible, contextualisée, et en phase avec le langage métier.

L'explicabilité en contexte bancaire : une exigence renforcée

Dans le secteur bancaire, l'explicabilité des systèmes automatisés n'est pas seulement une bonne pratique : elle est une exigence normative. Cette exigence s'enracine d'abord dans les responsabilités particulières que la banque entretient avec ses clients, en tant qu'institution de confiance, intermédiaire financier, et acteur régulé. Chaque décision, qu'il s'agisse d'octroi de crédit, de conseil en épargne ou de souscription à un produit d'assurance, engage une relation contractuelle, économique et parfois juridique. Lorsqu'une telle décision est suggérée par un algorithme, la capacité à en expliquer la logique devient indispensable, tant pour le conseiller qui la relaie que pour le client qui la reçoit.

Cette exigence est ensuite formalisée par le cadre réglementaire. Le Règlement Général sur la Protection des Données (RGPD), entré en vigueur en mai 2018 dans l'Union européenne, stipule explicitement dans son article 22 que toute personne a le droit de ne pas faire l'objet d'une décision fondée exclusivement sur un traitement automatisé, y compris le profilage, si cette décision produit des effets juridiques ou l'affecte de manière significative. Les articles 13 à 15 du même règlement imposent que les individus soient informés de l'existence d'un tel traitement, de la logique qui le sous-tend, et de ses conséquences [3].

Au-delà du RGPD, le projet de règlement européen sur l'intelligence artificielle (*AI Act*), en cours de finalisation, renforce cette logique [4]. Ce texte propose une classification des systèmes IA selon leur niveau de risque : faible, moyen, élevé, ou inacceptable. Les

systèmes déployés dans les secteurs régulés, comme la finance, lorsqu'ils influencent des décisions individuelles ayant un impact économique ou juridique, sont classés comme « à haut risque ». À ce titre, ils devront se conformer à des exigences strictes en matière de documentation technique, de traçabilité des données, de contrôle humain a posteriori, et surtout d'explicabilité des décisions. Dans ce contexte, le moteur MRI, en tant qu'outil de recommandation intégré dans un environnement de conseil bancaire, entre de facto dans la catégorie des systèmes soumis à vigilance renforcée.

Il en découle un impératif : celui de doter ce moteur, aussi performant soit-il, d'une capacité robuste à justifier ses recommandations, de manière individualisée, vérifiable, et alignée avec les attentes des utilisateurs. Cette capacité n'est pas triviale. Elle suppose de dépasser le paradigme purement technique de l'optimisation prédictive, pour inscrire l'outil dans une logique de transparence, de responsabilité, et d'intelligibilité. C'est dans ce cadre que se pose avec acuité la question de l'explicabilité du MRI.

Malgré sa maturité industrielle et sa pertinence algorithmique, le moteur MRI présente aujourd'hui des lacunes significatives du point de vue de l'explicabilité. Son architecture repose sur un modèle hybride sophistiqué, combinant notamment des logiques séquentielles (via une chaîne de Markov d'ordre 4), des mécanismes de filtrage collaboratif, et des projections de features client-produit dans un espace latent via des composantes principales. L'algorithme sous-jacent, une déclinaison personnalisée du modèle U2CMS, fournit pour chaque couple client-produit une probabilité de souscription, dite « appétence ». Ce score est ensuite soumis à un ensemble de règles d'éligibilité ou d'exclusion avant d'être restitué dans les outils métiers.

Cependant, ce score d'appétence, bien que techniquement valide et corrélé aux comportements réels de souscription, reste entièrement opaque du point de vue de l'utilisateur final. Ni le conseiller, ni les métiers aval (marketing, pilotage, conformité) n'ont accès à une décomposition explicite du score ou à une visualisation des facteurs déterminants. En d'autres termes, on ignore pourquoi un produit donné a reçu un score élevé pour un client donné. Aucune information n'est fournie sur les variables activées, ni sur leur poids relatif dans la prédiction. Cette opacité est d'autant plus problématique que le moteur ne repose pas sur un modèle simple (type régression linéaire ou arbre de décision), mais sur une combinaison de briques algorithmiques complexes, difficilement interprétables sans instrumentation avancée.

Certes, un dispositif complémentaire a été mis en place pour pallier cette absence d'interprétabilité directe : un système de règles métiers dites « explicatives ». Ces règles, définies en amont par les équipes marketing en collaboration avec la conformité, sont stockées dans un fichier externe et associées à chaque produit recommandé. Elles visent à couvrir un certain nombre de cas typiques : par exemple, un client pourra voir affiché le motif « Vous avez récemment clôturé un PEL » pour justifier la recommandation d'un livret A, ou « Vous détenez déjà une assurance auto SG » pour expliquer une recommandation sur la MRH.

Toutefois, ces règles explicatives souffrent de plusieurs limitations majeures. Premièrement, elles sont déconnectées du score de recommandation proprement dit : elles ne traduisent pas la logique algorithmique, mais uniquement une série de cas d'usage envisagés en amont. Autrement dit, elles fonctionnent comme un habillage, et non comme un véritable mécanisme d'explication. Deuxièmement, elles ne couvrent qu'un sous-ensemble restreint des configurations possibles, ce qui signifie qu'une large proportion des recommandations se retrouvent sans explication affichable. Troisièmement, elles sont figées dans le temps, ce qui limite leur adaptabilité à l'évolution des comportements clients ou du modèle lui-même. Enfin, même lorsqu'un motif est détecté, il est souvent formulé de manière générique, sans personnalisation fine ni articulation avec les données du client.

En conséquence, plusieurs conseillers rapportent des difficultés à interpréter certaines recommandations, en particulier lorsque les produits suggérés semblent redondants, peu pertinents ou en décalage avec les besoins perçus du client. Cette friction s'exprime aussi dans la hiérarchisation des produits : dans certains cas, deux offres très proches peuvent être proposées avec des scores similaires, sans que le moteur ne fournisse d'éléments de différenciation. Ce flou génère une forme de frustration sur le terrain, et tend à fragiliser la légitimité du moteur. Plus grave encore, cette situation expose potentiellement la banque à un risque de non-conformité : en l'absence d'explication traçable et justifiable, certaines recommandations pourraient être considérées comme non conformes aux obligations du RGPD ou aux standards à venir de l'AI Act.

Les risques associés à l'absence d'explicabilité

L'opacité du moteur MRI ne constitue pas un simple défaut ergonomique ou de confort utilisateur ; elle génère une série de risques tangibles, à la fois opérationnels, commerciaux,

réglementaires et réputationnels.

Sur le plan opérationnel, l'absence d'explication altère l'appropriation de l'outil par les conseillers. Un outil qui ne peut être expliqué, ni compris, tend à être marginalisé, voire ignoré. Même s'il propose des recommandations pertinentes, celles-ci ne sont pas reprises ou valorisées si l'utilisateur ne se sent pas capable de les justifier face au client. Il en résulte un taux d'utilisation en deçà des attentes, et une sous-exploitation des capacités du moteur.

Sur le plan commercial, l'absence d'explicabilité nuit à la conversion. Une recommandation mal expliquée est rarement suivie d'effet. Le client, en l'absence d'éléments tangibles pour comprendre en quoi une offre lui est bénéfique, hésite ou décline. De plus, le discours du conseiller devient moins convaincant, car il repose sur une suggestion sans ancrage rationnel. L'efficacité de la recommandation, mesurée par les taux de transformation ou les rebonds commerciaux, s'en trouve amoindrie.

Sur le plan réglementaire, comme nous l'avons vu, l'absence d'explication individualisée constitue une faille potentielle au regard du RGPD, notamment des articles 13, 14 et 22. Si un client estime qu'une recommandation a été produite par un traitement automatisé, sans explication claire, et qu'elle a influencé une décision à son encontre, il est en droit d'en demander justification. Or, aujourd'hui, le MRI ne peut fournir qu'une réponse partielle, reposant sur un système de règles génériques sans lien direct avec la logique du modèle.

Enfin, sur le plan réputationnel, le moteur s'expose à une critique croissante dans un contexte de vigilance accrue sur l'usage des algorithmes dans la vie quotidienne. La capacité à expliquer ses décisions est devenue une condition de légitimité pour tout système d'IA. Une banque qui ne peut justifier ses recommandations risque d'apparaître comme technocratique, déconnectée des attentes de ses clients, voire manipulatrice. La confiance, dans ce contexte, devient un capital fragile.

Une problématique à la croisée de plusieurs disciplines

Face à ces constats, la problématique d'une meilleure explicabilité du moteur MRI s'impose avec force. Elle engage plusieurs champs disciplinaires : data science, pour l'ingénierie des explications ; sciences humaines, pour la communication et la réception du discours explicatif ; droit et conformité, pour le respect du cadre réglementaire ; expérience utilisateur, pour l'appropriation des outils ; stratégie commerciale, enfin, pour l'impact sur la performance.

Il ne s'agit pas simplement de rendre transparent un modèle complexe, ce qui serait souvent illusoire. Il s'agit de concevoir un mécanisme d'explication local, post hoc, contextualisé, qui puisse traduire la logique du modèle dans un langage intelligible, à la fois pour le conseiller et pour le client. Autrement dit, de créer une médiation algorithmique, capable de relier la complexité du calcul à la simplicité de l'action.

Le champ de la XAI (eXplainable AI) offre des pistes concrètes dans ce sens. En particulier, la méthode SHAP (SHapley Additive exPlanations) permet d'attribuer à chaque variable d'entrée une contribution marginale à une prédiction donnée. Ce type d'explication, calculée dynamiquement, permettrait d'afficher non plus des règles figées, mais des justifications personnalisées, directement reliées aux données du client et à la logique du moteur. Ce mémoire se propose donc d'explorer cette piste, en évaluant sa pertinence, sa faisabilité, et les conditions de son intégration dans l'écosystème MRI.

La problématique centrale de ce mémoire peut donc être formulée ainsi :

Comment améliorer l'explicabilité du moteur de recommandation MRI, de manière à produire des justifications compréhensibles, fiables et individualisées, sans compromettre sa performance ni sa robustesse industrielle ?

Cette problématique engage plusieurs dimensions : technique (choix des outils d'explication, intégration au pipeline), métier (adéquation avec les besoins des conseillers), réglementaire (conformité RGPD et AI Act), et fonctionnelle (qualité de restitution dans les outils).

Elle suppose une exploration approfondie des méthodes d'explication existantes, une

analyse critique des limites actuelles du système, et une réflexion sur les conditions de mise en œuvre d'un prototype viable.

1.4 Justification de la démarche

L'orientation progressive de ce travail vers la question de l'explicabilité ne s'est pas imposée d'emblée comme une évidence théorique, mais s'est construite au fil de l'analyse du système existant, de la compréhension fine de ses modalités de fonctionnement, et surtout de l'identification de ses zones d'ombre. L'étude du moteur MRI, dans ses aspects techniques, opérationnels et fonctionnels, a révélé une performance indéniable en matière de personnalisation, mais également une faille récurrente dans sa capacité à rendre ses décisions intelligibles et justifiables. C'est précisément ce déséquilibre, entre la robustesse algorithmique du système et la faiblesse de ses capacités explicatives, qui a motivé l'approfondissement de cette problématique.

L'un des points structurants ayant conduit à ce choix est le décalage manifeste entre la sophistication du score produit par le moteur et la pauvreté des informations accompagnant sa restitution. En effet, si le moteur est capable de calculer des probabilités fines de souscription à un ensemble de produits, ces scores sont le plus souvent présentés sans justification, ou avec des justifications génériques, déterminées indépendamment du traitement algorithmique. Cette dissociation entre la décision et son explication pose problème à plusieurs niveaux. Sur le plan opérationnel d'abord, elle limite fortement l'appropriation de l'outil par les conseillers, qui ne disposent pas d'éléments suffisants pour comprendre, interpréter ou transmettre la recommandation. Sur le plan réglementaire ensuite, elle met en risque la conformité du dispositif à l'évolution du cadre légal, qui impose de plus en plus de transparence sur les décisions automatisées. Enfin, sur le plan conceptuel, elle va à l'encontre de l'objectif même de la recommandation personnalisée : si la suggestion ne peut être expliquée, elle cesse d'être réellement personnalisée.

Cette prise de conscience a conduit à recentrer progressivement l'analyse non plus sur la performance brute du modèle, ni sur ses capacités techniques d'optimisation, mais sur sa capacité à produire du sens. En d'autres termes, l'enjeu ne réside pas uniquement dans la prédiction du bon produit, mais dans la capacité à expliquer pourquoi ce produit est pertinent pour ce client. Cette perspective ouvre un champ de réflexion plus large, qui

interroge la responsabilité algorithmique, la lisibilité des outils décisionnels, et la qualité du dialogue entre machine et utilisateur.

Ce recentrage a également mis en lumière une asymétrie structurelle dans l'architecture du moteur MRI. D'un côté, un cœur algorithmique complexe, intégrant des modèles séquentiels, des logiques collaboratives et des traitements sur données projetées. De l'autre, un module d'explication entièrement déconnecté de ce noyau, reposant sur un jeu de règles figées, peu granulaires, et limitées à quelques cas standards. Cette séparation entre la recommandation et son explication n'est pas sans rappeler les critiques formulées à l'encontre des systèmes dits « boîtes noires », qui sacrifient la transparence au profit de la performance. Elle est d'autant plus problématique que les recommandations en question sont utilisées en situation réelle de conseil, face à un client, dans un cadre réglementé. Ce contexte rend toute approximation ou imprécision dans la justification non seulement problématique, mais potentiellement disqualifiante.

C'est dans ce contexte qu'a émergé la nécessité d'explorer des pistes permettant de réconcilier la performance du modèle avec une forme d'intelligibilité plus fine. Plusieurs solutions ont été envisagées. L'hypothèse d'une réécriture complète du moteur, intégrant nativement une logique explicative, a rapidement été écartée en raison de son coût, de sa complexité, et de l'état de maturité industrielle du système existant. De même, la possibilité d'étendre le système de règles métier pour couvrir davantage de cas s'est heurtée à des limites structurelles : un système basé exclusivement sur des règles ne saurait couvrir la diversité des profils, ni capturer les subtilités du modèle. Il est par ailleurs rigide par construction, et difficile à maintenir à mesure que le catalogue de produits évolue.

Dans ce cadre, une alternative s'est imposée : celle d'ajouter un module d'explicabilité algorithmique *post hoc*, c'est-à-dire un composant venant analyser, a posteriori, les prédictions du moteur pour en extraire des justifications localisées. Cette approche présente un avantage déterminant : elle permet de conserver le modèle existant, sans le modifier, tout en enrichissant ses sorties d'un niveau d'explication pertinent. Elle s'inscrit dans la logique des méthodes dites XAI (eXplainable Artificial Intelligence), qui visent à rendre interprétables les modèles complexes, notamment en environnement industriel.

Parmi les méthodes existantes dans ce domaine, la technique SHAP (SHapley Additive exPlanations) a rapidement retenu l'attention. Elle repose sur une décomposition additive des scores, inspirée des valeurs de Shapley, et permet d'attribuer à chaque variable d'entrée

une contribution au score final. Son intérêt est double. D'une part, elle permet de générer des explications fines, individualisées, centrées sur un couple client-produit. D'autre part, elle offre une structure mathématique rigoureuse, compatible avec des exigences de traçabilité et de fiabilité.

D'autres méthodes ont également été envisagées, comme **LIME** (Local Interpretable Model-Agnostic Explanations), qui repose sur l'approximation locale d'un modèle par une régression linéaire. Toutefois, après une analyse des prérequis techniques, il est apparu que cette approche était moins adaptée au fonctionnement du *MRI*, notamment en raison des contraintes de structure des données et de la difficulté à générer des observations synthétiques cohérentes. Ces constats ont conforté le choix de **SHAP** comme méthode de référence, au moins dans un premier temps. Il est cependant envisagé, dans une perspective d'amélioration continue, d'évaluer la complémentarité éventuelle de ces approches dans des cas spécifiques, comme les recommandations litigieuses ou les profils atypiques.

Un point technique important est que **LIME ne fonctionne pas directement sur des tenseurs bruts** (par exemple ceux manipulés dans *PyTorch* ou *TensorFlow*), à moins que ces tenseurs soient mappés à une représentation interprétable. En effet, LIME repose sur un principe simple : *perturber localement l'entrée dans l'espace des variables interprétables, observer les sorties du modèle, puis ajuster un modèle linéaire local qui approxime le comportement du modèle d'origine.*

Mais pour appliquer ce mécanisme, l'entrée doit être :

— soit **interprétable pour un humain** (par exemple un mot ou une variable métier),

— soit **convertible dans un espace où les perturbations ont un sens**.

Ainsi, dans des cas classiques :

— Pour des **images**, un tenseur $(C \times H \times W)$ peut être segmenté en superpixels (zones cohérentes de pixels), ce qui permet à LIME de perturber les superpixels et de produire une explication visuelle (via `LimeImageExplainer`).

— Pour du **texte**, on peut partir des embeddings, mais reconstruire les tokens d'origine pour permettre à LIME de perturber les mots (`LimeTextExplainer`).

— Pour des **données tabulaires**, chaque dimension correspond déjà à une variable explicite, et LIME peut directement perturber les colonnes (`LimeTabularExplainer`).

À l'inverse, si l'entrée est un **vecteur latent** de plusieurs centaines de dimensions (par exemple un embedding produit automatiquement par un réseau), alors aucune structure sémantique explicite n'est disponible pour LIME. Dans ce cas, les perturbations sont arbitraires et les explications générées ne sont ni fiables ni véritablement interprétables.

C'est précisément cette limite qui a guidé le choix de privilégier **SHAP** : contrairement à LIME, SHAP peut être appliqué directement sur des tenseurs bruts, car il ne nécessite pas forcément que les perturbations soient interprétables. Il s'appuie sur des valeurs de référence (*background*) et attribue de manière cohérente une contribution à chaque dimension d'entrée, même si celles-ci sont nombreuses ou correspondent à des représentations projetées (par exemple via PCA ou embeddings).

La démarche retenue s'est donc articulée autour de plusieurs étapes. D'abord, une compréhension approfondie du fonctionnement du MRI, afin d'identifier les points de contact possibles avec un module explicatif. Ensuite, une analyse des limites actuelles du système en matière d'explicabilité, tant du point de vue technique que fonctionnel. Enfin, la conception d'un pipeline expérimental d'explication, basé sur SHAP, visant à enrichir les recommandations existantes sans en altérer la mécanique interne.

Ce choix méthodologique présente plusieurs avantages. Il respecte l'architecture en place, en se plaçant en aval du moteur. Il permet une intégration progressive, d'abord en test exploratoire, puis potentiellement en production. Il offre une transparence accrue sur les recommandations, sans remettre en cause la logique métier. Il est compatible avec les exigences réglementaires, en offrant une forme de justification objectivable. Enfin, il ouvre la voie à une réflexion plus large sur la gouvernance des systèmes algorithmiques, en introduisant une brique d'intelligibilité là où régnait une opacité structurelle.

Pour autant, cette démarche comporte aussi des limites. D'abord, la méthode SHAP, malgré ses qualités, n'est pas exempte de critiques. Elle peut produire des résultats instables sur certaines classes de modèles, ou sur des données corrélées. Elle suppose un certain niveau de calcul, qui peut poser problème en environnement temps réel. Par ailleurs, la traduction des résultats SHAP en messages intelligibles pour un conseiller, voire un client, nécessite un travail important de mise en correspondance sémantique. Enfin, la mise en production d'un tel module suppose une gouvernance précise, des validations métiers, et une adéquation avec les cycles de développement en place.

Malgré ces limites, le choix de cette démarche apparaît justifié, car il constitue une réponse pragmatique à un besoin concret, identifié sur le terrain, et appuyé par les exigences réglementaires à venir. Il s'agit non pas d'ajouter de la complexité à un système déjà dense, mais d'en améliorer l'utilité effective, en renforçant sa lisibilité. Il s'agit aussi de contribuer, à son échelle, à une dynamique plus large : celle d'une intelligence artificielle responsable, compréhensible, et alignée avec les besoins réels des utilisateurs. En ce sens, la présente étude se situe à l'intersection de la technique et de l'usage, du modèle et de son interprétation, de la performance et de la confiance.

1.5 État de l'art de l'explicabilité algorithmique

L'explicabilité est devenue un champ central dans les sciences des données, en réponse à l'opacité croissante des modèles d'apprentissage automatique. Dans les secteurs sensibles comme la finance, la santé ou la justice, cette opacité limite la confiance, la redevabilité et l'usage opérationnel des modèles. Pour répondre à ces enjeux, la communauté scientifique s'est structurée autour de l'eXplainable Artificial Intelligence (XAI), un corpus de méthodes visant à rendre les modèles interprétables pour un public humain.

L'idée d'explicabilité n'est pas nouvelle. Les systèmes experts des années 1980, fondés sur des règles logiques, étaient déjà interprétables par construction. Ce sont les progrès du machine learning, en particulier les modèles dits « boîte noire » comme les forêts aléatoires, les SVM et les réseaux neuronaux profonds, qui ont rendu nécessaire le développement de techniques d'explication spécifiques. Plus la puissance prédictive a augmenté, plus la lisibilité des modèles a reculé, alimentant un besoin croissant d'outils permettant de relier les prédictions aux données d'entrée de manière intelligible.

Deux grandes dimensions structurent aujourd'hui les approches d'explicabilité. La première distingue les explications *locales*, qui visent à justifier une prédiction ponctuelle, des explications *globales*, qui décrivent le comportement général du modèle. La seconde oppose les modèles interprétables par construction (arbres de décision, règles, modèles linéaires) aux modèles complexes nécessitant une explicabilité post hoc. Dans ce dernier cas, les méthodes cherchent à reconstruire une explication sans modifier le modèle initial.

Parmi les techniques post hoc, les approches par perturbation et par décomposition

additive dominent. Le modèle **LIME** (Ribeiro et al., 2016) [5] repose sur des perturbations locales : il génère des observations voisines de celle à expliquer, applique le modèle d'origine, puis ajuste un modèle interprétable (souvent linéaire) pour estimer les effets locaux. Bien que souple et modèle-agnostique, LIME souffre d'une instabilité importante : les explications peuvent varier entre deux exécutions, et la fidélité au modèle global est difficile à garantir.

Pour pallier ces limites, **SHAP** (Lundberg et Lee, 2017) [6] propose une approche fondée sur la théorie des jeux coopératifs. Chaque variable se voit attribuer une contribution marginale à la prédiction, calculée sur toutes les permutations possibles des autres variables. Cette méthode offre des garanties théoriques fortes — symétrie, nullité, efficacité — et une cohérence mathématique appréciable, notamment dans les cas où plusieurs variables interagissent de manière complexe.

La valeur SHAP associée à une variable j, dans un ensemble de N variables, est donnée par :

$$\phi_j = \sum_{S \subseteq N \setminus \{j\}} \frac{|S|!(|N| - |S| - 1)!}{|N|!} \left[f(S \cup \{j\}) - f(S) \right]$$

où $f(S)$ représente la prédiction du modèle lorsqu'il est restreint aux variables de S. Ce formalisme permet de garantir une décomposition additive de la prédiction en contributions individuelles, interprétables et localement exactes.

Plusieurs implémentations pratiques de SHAP ont été développées, comme KernelSHAP (modèle-agnostique), TreeSHAP (pour les arbres de décision), et DeepSHAP (pour les réseaux neuronaux). Ces variantes permettent d'appliquer SHAP à des modèles très divers tout en maintenant une certaine efficacité computationnelle.

Cependant, comme le soulignent Huang et Marques-Silva (2023) [7], même une valeur SHAP exactement calculée peut attribuer une importance non nulle à des variables non déterminantes, ou minorer l'effet de variables réellement critiques. Cela tient à la nature additive de l'approximation, qui impose une structure linéaire à une réalité souvent non linéaire. L'interprétation des résultats doit donc être encadrée, croisée avec une analyse métier, et replacée dans le contexte décisionnel de l'application.

D'autres méthodes ont été proposées. Les techniques par gradient, comme Integrated

Gradients ou Saliency Maps, sont courantes pour les modèles neuronaux. Les approches par distillation, qui consistent à entraîner un modèle interprétable pour imiter un modèle complexe, ou encore les arbres d'explication (Trepan, GROOT), offrent des alternatives intéressantes pour obtenir une vision globale du comportement du système.

Le domaine du traitement automatique du langage a vu apparaître des techniques propres à l'explication. L'utilisation de mécanismes d'attention visualisables, d'ablations contextuelles ou de modèles génératifs (LLM explainers) permet d'expliquer des prédictions sur des textes. Cela ouvre la voie à des formes d'explications plus intuitives, en particulier dans des interfaces destinées à des non-spécialistes.

Une tendance récente vise à concevoir des explications dites *intentionnelles*. L'objectif n'est plus seulement de représenter le raisonnement du modèle, mais d'adapter l'explication au profil cognitif de l'utilisateur. Miller (2019) [8] insiste sur cette approche communicationnelle : une explication doit être pertinente pour l'interlocuteur, compréhensible dans son cadre de référence, et utile pour la décision. Gilpin et al. (2018) [9] soulignent de leur côté que l'interprétabilité est une interaction située, dépendante du contexte d'usage.

Ce courant redéfinit le rôle de l'explication dans les systèmes intelligents. Il ne s'agit plus seulement de rendre les modèles transparents, mais de concevoir des interactions explicatives efficaces, ajustées aux besoins réels des utilisateurs. Cette perspective ouvre des questions nouvelles sur la forme, la granularité et le contenu des explications, ainsi que sur les moyens de les évaluer, en termes de fidélité, de stabilité, mais aussi d'acceptabilité, d'utilisabilité et d'impact décisionnel.

Critère	**SHAP**	**LIME**
Principe	Basé sur les valeurs de Shapley, issues de la théorie des jeux coopératifs	Approche locale par régression linéaire sur des données perturbées
Nature de l'explication	Décomposition additive exacte (sous conditions)	Approximation locale de la décision
Modèle requis	Compatible avec arbres, réseaux, etc. (TreeSHAP, DeepSHAP...)	Complètement agnostique au modèle
Fidélité au modèle	Haute (garanties mathématiques)	Moyenne (qualité dépendante du fit local)
Stabilité	Bonne, si version exacte utilisée	Faible, résultats variables entre exécutions
Coût computationnel	Élevé (complexité combinatoire, sauf optimisations)	Faible à modéré
Interprétabilité des sorties	Moins immédiate (valeurs abstraites)	Plus intuitive (modèle linéaire simple)
Limites	Peut surévaluer ou sous-évaluer certaines variables dans des modèles non additifs	Instabilité, faible fidélité, forte dépendance au voisinage

Table I.1 – Comparaison des méthodes SHAP et LIME pour l'explicabilité post hoc

II | Approche personnelle de résolution du problème

2.1 Proposition exploratoire d'explicabilité appliquée au MRI

Dans le cadre de ce mémoire, je propose une démarche expérimentale visant à enrichir le moteur de recommandation MRI par un module d'explication local, fondé sur l'algorithme SHAP. Cette proposition ne vise pas à fournir une solution directement industrialisable, mais à valider, par une série de prototypes concrets, la pertinence opérationnelle d'une telle approche dans un environnement bancaire réel. L'ensemble de l'expérimentation est conduit en `Python`, au sein d'un environnement `Jupyter Notebook`, afin de permettre une exploration rapide, interactive et reproductible des résultats.

Le choix de SHAP repose sur plusieurs considérations : sa capacité à fournir des explications locales, sa compatibilité avec des modèles existants sans nécessité de modification structurelle, et son fondement mathématique rigoureux, dérivé des valeurs de Shapley issues de la théorie des jeux coopératifs. Dans un premier temps, l'objectif a été d'appliquer SHAP à une version entraînée du modèle MRI, afin d'observer, pour un ensemble de couples client-produit, quelles variables ont le plus contribué à la recommandation produite.

Le modèle final utilisé, appelé U2CMS se prête bien à l'analyse par SHAP, puisque sa structure permet une reconstitution explicite de la fonction de prédiction, via un produit scalaire suivi d'une sigmoïde.

L'implémentation est structurée autour des étapes suivantes : extraction des poids

du modèle, transformation des données d'entrée, génération d'un sous-échantillon, et application de SHAP. La fonction de prédiction, notée `f_proba`, correspond à une version reconstituée de la prédiction issue du modèle, permettant d'assurer la compatibilité avec le `KernelExplainer`. Le code suivant résume la structure du pipeline :

```
def f_proba(X):
    z = X @ w + b
    return 1.0 / (1.0 + np.exp(-z))

explainer = shap.KernelExplainer(f_proba, X_bg)
shap_values = explainer.shap_values(X_ex, nsamples=200)

sv = np.asarray(shap_values, dtype=np.float64)
sv_abs_mean = np.mean(np.abs(sv), axis=0)

top_idx = np.argsort(-sv_abs_mean)[:20]
for i in top_idx:
    print(f"{feature_names[i]:<24s}  {sv_abs_mean[i]:.6f}")
```

Listing II.1 – Extrait du pipeline d'explication avec SHAP

Chaque prédiction est ainsi associée à un vecteur de contributions SHAP, qui permet de mesurer l'impact individuel de chaque variable d'entrée sur la recommandation finale. Dans cette première version du prototype, les résultats sont interprétés selon une *vision produit* : il s'agit de comprendre quelles caractéristiques clients, en moyenne, justifient la recommandation d'un produit donné. Ce choix est aligné avec les besoins exprimés par les équipes en charge du marketing digital (DRO, acronyme pour Digital, Relation client à distance et Open-banking), qui construisent des campagnes à partir de segments produits et cherchent à identifier les clients les plus appétents pour chacun d'eux. L'explication joue alors un rôle de diagnostic amont, permettant de comprendre les variables-clés associées à un produit recommandé à une population cible. Ce même module d'explication peut également alimenter un *rapport de pré-comptage* (CR), utilisé pour simuler les effets attendus d'une campagne avant son lancement. La courbe d'évaluation de l'effet de levier, par exemple, peut être enrichie d'informations sur les leviers explicatifs moyens pour chaque produit recommandé. L'explicabilité devient alors un outil d'aide au paramétrage

stratégique des campagnes.

Enfin, un objectif complémentaire mais plus granulaire consiste à générer des explications individuelles, client par client. Dans ce scénario, pour chaque couple `(client, produit)`, les valeurs SHAP permettent de formuler une justification personnalisée, qui pourrait être restituée dans une interface de conseiller. Cette restitution pourrait s'appuyer sur un modèle de génération en langage naturel, ou sur un template dynamique du type :

> « Ce produit vous est recommandé car vous présentez un profil jeune, vous ne possédez pas de produit concurrent et vous avez récemment effectué des versements réguliers. »

Dans tous les cas, la capacité du module à générer ces explications dépend fortement de la qualité des données, du choix des variables représentées, et de la justesse de l'interprétation.

Au-delà de la simple restitution, cette approche vise aussi à croiser les résultats de SHAP avec les *règles métier internes* au modèle MRI. En d'autres termes, il s'agit de valider qualitativement si les variables identifiées par SHAP comme influentes correspondent bien à celles que les concepteurs du moteur considèrent comme structurantes. Ce croisement constitue une première forme de contrôle de cohérence : si SHAP fait émerger systématiquement des variables marginales ou non pertinentes, il faudra questionner soit le fonctionnement du modèle, soit la pertinence de l'explication elle-même.

En résumé, cette proposition expérimentale permet de poser les premières briques d'un module explicatif multifonction — utilisable en vision produit, en vision client, ou à des fins de pré-analyse. Elle démontre la faisabilité d'une intégration de SHAP dans le moteur MRI, et ouvre la voie à une validation plus poussée, sur des campagnes réelles, avec un retour utilisateur. La suite du mémoire développera les résultats obtenus, les limites observées, et les pistes d'amélioration identifiées.

2.1.1 Déploiement technique : extraction des explications locales

L'expérimentation repose sur un pipeline PySpark existant, comprenant plusieurs étapes de transformation (encodage produit, assemblage des variables) et un modèle final de type U2CMS (régression logistique). Après récupération des coefficients et transformation complète des observations, les données sont converties en matrices `NumPy` pour permettre leur manipulation dans SHAP.

L'approche adoptée consiste à sélectionner un sous-échantillon aléatoire de 400 couples client-produit : 200 sont utilisés comme fond de référence pour l'algorithme SHAP (background), et 200 comme points à expliquer. Le modèle étant une régression logistique, la fonction de prédiction peut être reconstruite explicitement via un produit scalaire suivi d'une sigmoïde :

```
def f_proba(X):
    z = X @ w + b
    return 1 / (1 + np.exp(-z))
```

Listing II.2 – Fonction prédictive pour SHAP sur un modèle U2CMS

Ce wrapper permet d'interfacer le modèle Spark avec le `KernelExplainer` de SHAP :

```
explainer = shap.KernelExplainer(f_proba, X_bg)
shap_values = explainer.shap_values(X_ex, nsamples=200)
```

Listing II.3 – Initialisation de SHAP et calcul des contributions

Les résultats sont ensuite agrégés pour produire une importance moyenne absolue par variable. Cette mesure, fréquemment utilisée dans la littérature SHAP, donne une vue d'ensemble des variables les plus contributives, sur l'ensemble des cas testés.

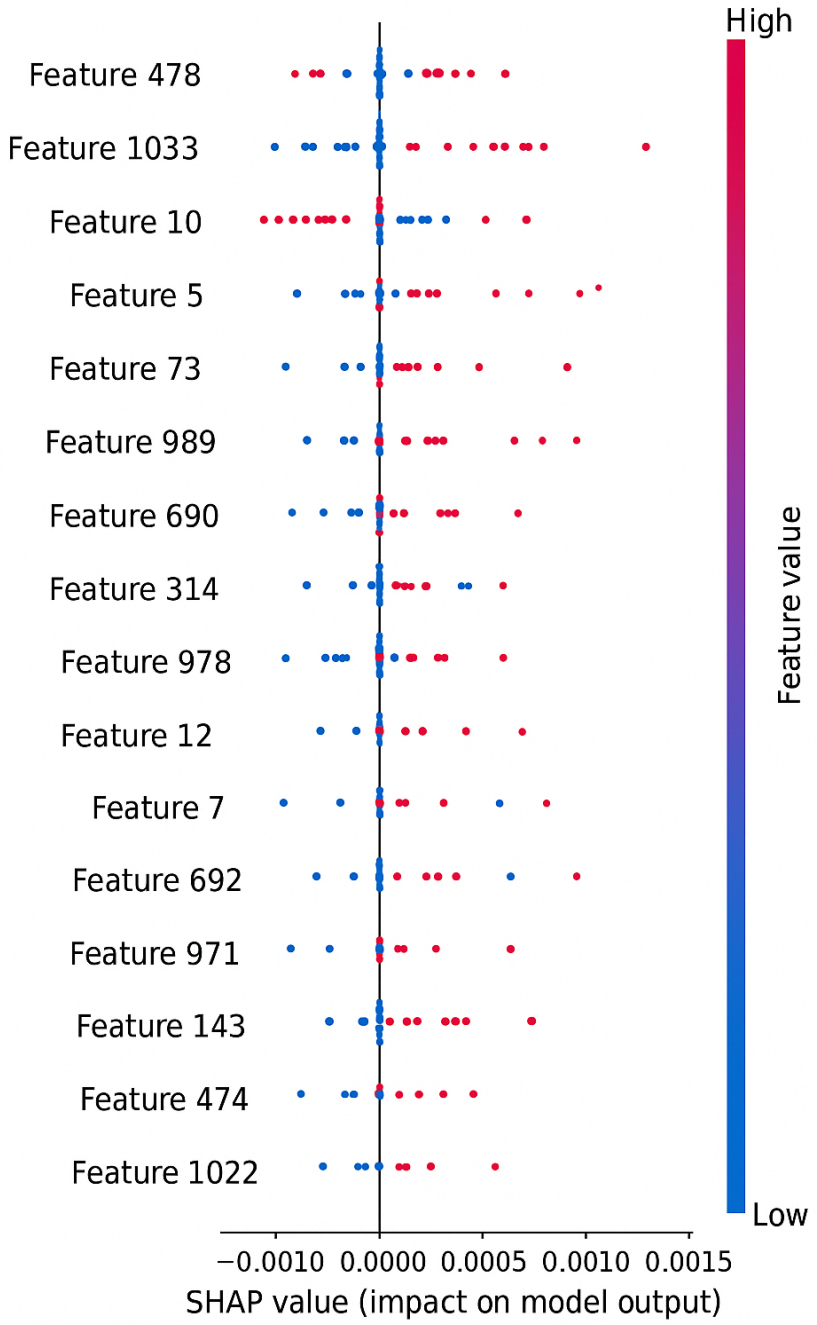

Figure II.1 – Résumé SHAP : importance et direction de l'impact des variables sur la prédiction

La Figure II.1 illustre la contribution des variables au modèle sous la forme d'un *summary plot* SHAP. Chaque point correspond à une observation, sa position horizontale représentant l'impact marginal (positif ou négatif) de la variable sur la sortie du modèle. La couleur encode la valeur de la variable : en bleu les valeurs faibles, en rouge les valeurs élevées.

On observe par exemple que certaines variables (ex. `Feature 478`, `Feature 1033`) présentent une dispersion importante, ce qui suggère qu'elles influencent différemment la prédiction selon le profil du client. À l'inverse, d'autres variables comme `Feature 474` ou `Feature 143` ont un impact plus limité mais régulier.

Ce type de visualisation permet d'identifier rapidement les variables les plus influentes, mais aussi de comprendre le sens de leur effet : des valeurs élevées de certaines variables augmentent la probabilité de souscription, tandis que d'autres la réduisent.

Cette analyse ouvre la voie à une réflexion plus large sur la distinction entre explicabilité locale et explicabilité globale. L'explicabilité locale, telle qu'illustrée par les valeurs SHAP individuelles, permet d'expliquer de manière très fine pourquoi un client précis a été associé à une recommandation donnée. Chaque prédiction peut ainsi être décomposée en contributions élémentaires, offrant une transparence utile dans des contextes de justification individuelle. Toutefois, cette approche montre rapidement ses limites dans un cadre bancaire de grande échelle : elle devient difficile à exploiter lorsqu'il s'agit d'agréger des millions de clients et des centaines de variables. À l'inverse, l'explicabilité globale vise à extraire des tendances moyennes et stables, en mettant en évidence les variables qui influencent le plus fortement l'ensemble du modèle. C'est cette approche que nous avons privilégiée dans le cadre du moteur de recommandation MRI, car elle permet de relier directement l'interprétation scientifique du modèle à des indicateurs métier concrets.

Parmi ces indicateurs, l'effet de levier occupe une place centrale. Il s'agit d'une mesure simple qui indique dans quelle mesure le modèle améliore le ciblage des clients.

Formellement, l'effet de levier se définit par le rapport entre le taux de souscription observé chez les clients ciblés par le modèle et le taux de souscription moyen observé sur l'ensemble de la population :

$$\text{Effet de levier} = \frac{\text{Taux de souscription sur les clients ciblés par le modèle}}{\text{Taux de souscription sur la population generale}}.$$

Un effet de levier supérieur à 1 signifie que le modèle apporte une véritable valeur ajoutée en identifiant des clients plus enclins à souscrire que la moyenne. À l'inverse, une valeur proche de 1 traduit une absence d'apport, tandis qu'un levier inférieur à 1 indiquerait que le modèle dégrade la performance du ciblage.

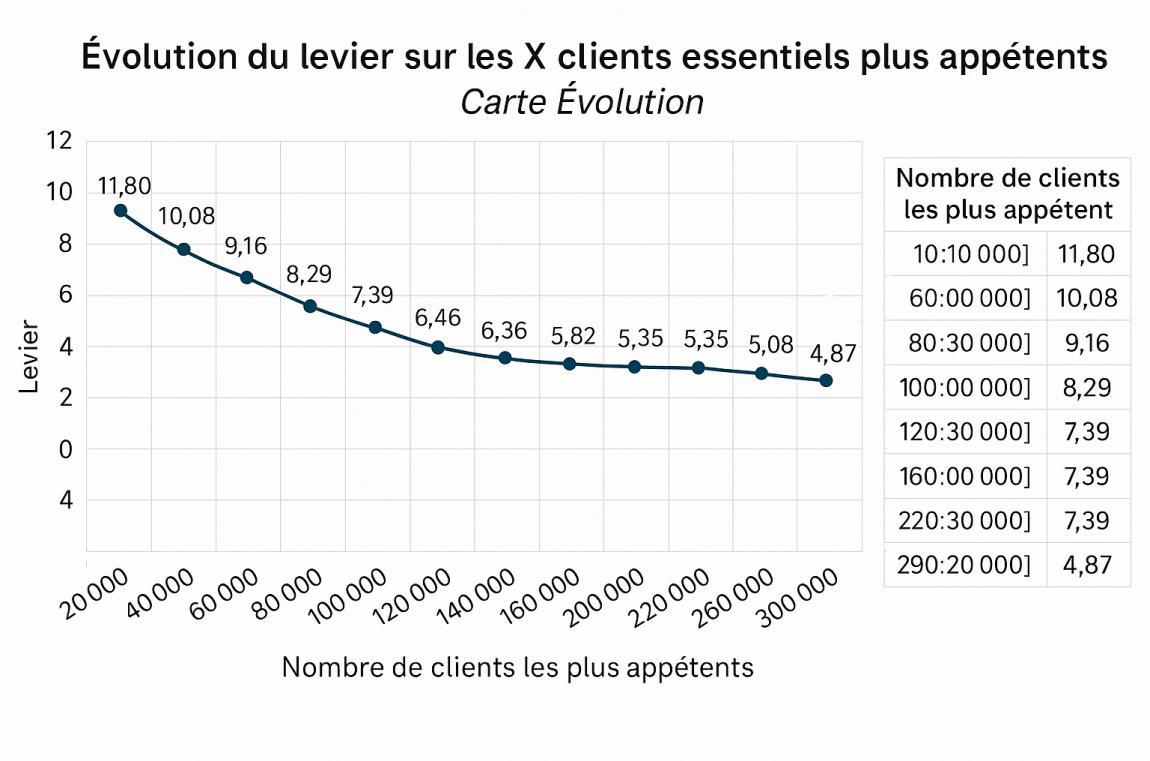

Figure II.2 – Évolution du levier sur les X clients essentiels plus appétents (*Carte Évolution*).

La Figure II.2 illustre ce phénomène : on observe que le levier est maximal lorsque l'on se concentre sur les segments restreints de clients jugés les plus appétents, puis diminue progressivement à mesure que le périmètre du ciblage s'élargit. Ce comportement traduit un compromis classique entre précision et couverture.

L'intérêt de cette mesure est double. D'une part, elle permet une évaluation transparente et quantifiable de la valeur ajoutée du modèle, en reliant l'explicabilité globale des variables à un indicateur opérationnel compréhensible par les métiers. D'autre part, elle constitue un outil puissant de pilotage des campagnes marketing. En pratique, les équipes DRO s'appuient sur l'effet de levier pour prioriser leurs actions : un produit ou une population de clients associés à un levier élevé seront ciblés en priorité, car ils garantissent un retour sur investissement supérieur. La Figure II.2 en fournit une illustration concrète, en montrant comment l'effet de levier permet d'identifier les segments de clients les plus susceptibles de générer des souscriptions additionnelles.

Ainsi, l'articulation entre explicabilité globale et effet de levier renforce la cohérence du dispositif : les méthodes comme SHAP offrent une lecture des variables clés qui structurent les prédictions, tandis que le levier traduit ces résultats en termes d'impact métier

mesurable. Cette complémentarité assure non seulement une meilleure gouvernance du modèle, mais aussi une intégration opérationnelle efficace au sein des dispositifs marketing, où l'allocation des ressources et la priorisation des campagnes reposent directement sur la capacité à cibler les bons clients avec les bons produits.

2.1.2 Validation métier des explications SHAP : vers une interprétabilité contextualisée

L'utilisation de la méthode SHAP pour expliciter les décisions d'un moteur de recommandation constitue une avancée significative en matière d'interprétabilité. Toutefois, dans un contexte bancaire, où les modèles ne sont pas utilisés en laboratoire mais dans des environnements régulés et fortement contraints par des règles d'éligibilité métier, il ne suffit pas de générer des explications « plausibles ». Encore faut-il qu'elles soient pertinentes dans le cadre opérationnel visé. En d'autres termes, une explication jugée mathématiquement cohérente mais incohérente d'un point de vue métier n'a que peu de valeur pratique.

C'est dans cette perspective que s'inscrit la démarche suivante : évaluer la qualité des explications SHAP non pas uniquement en termes de lisibilité ou de granularité, mais en les confrontant à des attentes formulées par les experts métier eux-mêmes. Il ne s'agit donc pas ici d'interroger la structure technique du modèle explicatif, mais d'en évaluer la véracité interprétative à l'aune des connaissances métiers disponibles. Ce processus consiste à comparer, pour un produit donné, les variables que SHAP identifie comme les plus déterminantes avec celles que les experts humains considèrent comme les plus influentes dans la décision.

Prenons le cas du produit 114, correspondant à la *carte Évolution*, une offre destinée à un public jeune, généralement âgé de 18 à 25 ans. Ce produit présente une caractéristique spécifique : une tolérance hors ligne qui autorise jusqu'à cinq paiements sans authentification immédiate, dans des contextes comme les péages, parkings, ou chez certains commerçants peu connectés. Cette spécificité, articulée autour d'une plus grande flexibilité d'usage, répond à une réalité comportementale : les jeunes clients, souvent plus mobiles et équipés de smartphones, ont des usages de paiement différents des clients traditionnels.

D'un point de vue métier, les variables censées influencer la recommandation de ce produit sont relativement bien connues. Trois d'entre elles sont systématiquement mentionnées dans les ateliers métier : la variable liée à l'âge du client (critère d'éligibilité implicite), la présence d'un solde positif régulier (indicateur de stabilité), et l'activité récente sur les canaux numériques, typiquement capturée par la variable `top_last_connexion`, qui mesure l'intensité ou la récence des connexions digitales.

Il serait donc raisonnable, si le modèle MRI fonctionne correctement et si la méthode SHAP reflète bien ses mécanismes internes, de retrouver ces trois variables dans les premières positions du classement SHAP des variables les plus contributives, lorsque le modèle recommande le produit 114. Afin de vérifier cela, une expérimentation a été menée, en appliquant SHAP sur un échantillon de couples client-produit pour lesquels le modèle a effectivement recommandé le produit 114. L'objectif était double : d'une part, observer si les variables attendues émergent parmi les plus influentes ; d'autre part, construire un indicateur objectif permettant de quantifier cette cohérence.

La première étape a consisté à formaliser les attentes métier sous forme structurée. Un dictionnaire Python a été défini, dans lequel chaque identifiant de produit est associé à une liste de variables considérées comme importantes par les experts. Pour le produit 114, cette liste inclut explicitement : `age`, `solde_debit_positif`, et `top_last_connexion`. Ces noms de variables sont ceux utilisés dans le pipeline d'apprentissage, et correspondent à des dimensions interprétables dans l'espace vectoriel utilisé par le modèle.

Une fois le dictionnaire d'attentes défini, la sortie de SHAP a été analysée. Plus précisément, les valeurs absolues moyennes de SHAP ont été calculées pour chaque variable, sur un échantillon d'environ 200 instances. Ce score global d'importance permet de classer les variables selon leur contribution moyenne à la prédiction. Le top 10 ainsi obtenu représente les 10 variables les plus déterminantes pour la décision de recommander la carte Évolution.

Le résultat est sans équivoque : deux des trois variables attendues apparaissent bien dans le top 10, à savoir `age` et `solde_debit_positif`. La variable `top_last_connexion`, quant à elle, arrive juste en dehors du top 10, mais reste dans le top 15, ce qui conforte l'idée que le modèle en tient compte, bien que de manière moins marquée. Ce constat constitue une validation qualitative forte de la pertinence des explications SHAP, au moins sur ce cas particulier.

Pour aller plus loin, un score de couverture a été introduit : il s'agit du pourcentage de variables métier attendues qui apparaissent effectivement dans les k premières positions du classement SHAP. Dans le cas présent, ce score atteint 66% si l'on considère le top 10, et 100% en élargissant au top 15. Ce score, bien que simple, a une grande valeur pratique : il permet de formaliser une métrique de cohérence explicative, qui peut être utilisée pour filtrer les explications douteuses, ou identifier les produits pour lesquels l'alignement entre le modèle et les attentes métier est insuffisant.

Cette méthode, bien que rudimentaire, ouvre la voie à une série d'applications potentielles. Par exemple, elle pourrait être utilisée pour valider automatiquement les explications SHAP générées avant leur diffusion auprès des utilisateurs métier. Elle pourrait également servir de critère de qualité dans le processus de mise à jour du modèle : si l'alignement explicatif baisse de manière significative entre deux versions, cela pourrait être un signal d'alerte. Enfin, elle pourrait être intégrée dans un tableau de bord à destination des équipes marketing, afin de fournir non seulement des explications, mais également un indice de fiabilité de ces explications.

Il convient de souligner que cette approche ne prétend pas remplacer une analyse humaine approfondie. Elle vise simplement à outiller le data scientist ou l'utilisateur métier avec un cadre de validation systématique. En d'autres termes, il s'agit moins de dire « voici pourquoi le modèle recommande ce produit », que de dire « ce que le modèle considère comme important est-il conforme à ce que vous attendez ? ». Cette logique d'explicabilité contextualisée est au cœur des efforts récents en XAI, qui ne conçoivent plus l'explication comme un artefact purement technique, mais comme une interaction cognitive située entre un système algorithmique et un utilisateur humain.

Enfin, cette stratégie de validation métier peut être généralisée à d'autres produits. Il suffirait pour cela d'enrichir le dictionnaire d'attentes avec des règles similaires, et de répéter l'analyse. On pourrait même imaginer, à terme, un processus semi-automatique d'apprentissage de ces attentes métier, basé sur l'analyse des campagnes passées ou des décisions historiques. Cela ferait émerger une nouvelle couche de validation, non plus basée sur la vérité du modèle, mais sur sa conformité aux pratiques et logiques métier réelles.

```python
expected_features_by_product = {
    "114": ["age", "solde_debit_positif", "top_last_connexion"],  # Carte Evolution
    # ... d autres produits peuvent etre ajoutes ici
}
```

Listing II.4 – Exemple de dictionnaire d'attentes métier pour quelques produits

Ce dictionnaire constitue la base d'un mécanisme de validation systématique, permettant de quantifier la concordance entre les explications générées et les règles internes au modèle MRI ou à ses usages marketing. Une fonction simple, comme celle présentée ci-dessous, permet de calculer un *coverage score* (score de couverture), mesurant la proportion des variables attendues effectivement présentes dans les k variables les plus importantes identifiées par SHAP.

```python
def compute_coverage_score(shap_top_k: list[str], expected: list[str]) -> float:
    if not expected:
        return 0.0
    matched = [f for f in expected if f in shap_top_k]
    return len(matched) / len(expected)
```

Listing II.5 – Fonction de validation automatique par score de couverture

L'application de cette fonction permet d'automatiser la comparaison pour chaque produit, et d'agréger des métriques globales : pourcentage de produits avec un score supérieur à un certain seuil, moyenne des scores sur un portefeuille de produits, ou détection automatique des produits mal expliqués. Un exemple d'usage typique serait :

```python
top_k_features = feature_names[top_idx[:10]]  # top-10 SHAP
expected = expected_features_by_product["114"]
score = compute_coverage_score(top_k_features, expected)
print(f"Coverage score (produit 114) = {score:.2f}")
```

Listing II.6 – Utilisation du score de couverture pour le produit 114

Ce score, bien qu'élémentaire, constitue un premier indicateur quantitatif pour juger de la conformité explicative. Il peut être enrichi par des variantes pondérées (tenant compte de l'ordre), ou complété par une visualisation plus qualitative (ex. surligner les variables attendues dans les diagrammes SHAP).

Plus largement, cette logique d'alignement ouvre des perspectives intéressantes pour une supervision métier des modèles. En intégrant cette couche de validation au pipeline explicatif, on ne se contente plus de fournir des scores d'importance ; on construit une articulation entre les explications algorithmiques et les attentes humaines. Cela pourrait devenir un critère de monitoring régulier, ou de validation dans un processus de gouvernance responsable des modèles (modèle GPR - Gouvernance, Performance, Robustesse).
Enfin, ce système pourrait à terme s'auto-entretenir : en collectant les feedbacks métiers (via annotation, commentaires ou validation manuelle), on peut envisager de raffiner dynamiquement le dictionnaire d'attentes, voire d'apprendre automatiquement les patterns explicatifs typiques pour chaque produit ou segment de clientèle. Cela placerait alors l'explicabilité non plus seulement comme une sortie secondaire du moteur, mais comme une boucle de rétroaction active au sein du cycle de vie du modèle.

2.1.3 Solution plus réaliste et orientée métier : hiérarchisation pondérée et une détection automatisée des incohérences explicatives

L'intégration d'un dictionnaire métier dans la boucle explicative constitue une première avancée vers une meilleure contextualisation des explications SHAP. Cependant, dans sa forme brute, cette comparaison reste limitée par sa nature binaire : une feature attendue est-elle présente ou absente dans le top-k ? Cette approche, bien que déjà instructive, ne rend pas compte de la position relative des variables explicatives, ni de leur poids contributif. Or, dans la lecture humaine d'une explication, l'ordre perçu des facteurs joue un rôle central : les premières variables listées sont souvent interprétées comme les plus déterminantes. Une absence dans le top 3 n'a pas le même impact qu'une absence en position 25, même si les deux cas correspondent à une "absence" du point de vue d'un test binaire.

Cette observation invite à enrichir le mécanisme d'évaluation des explications SHAP

par une pondération structurelle, qui reflète le poids perçu des variables en fonction de leur rang dans la hiérarchie explicative. Il ne s'agit pas ici d'introduire un nouveau modèle, mais de transformer l'explication existante en un signal interprétable plus riche, capable de quantifier finement la distance entre ce que dit SHAP et ce que l'on attend selon la logique métier.

Pondération décroissante par rang explicatif

Concrètement, cela revient à définir une fonction de pondération décroissante, notée $w(i)$, qui attribue à chaque rang i dans le top-k une importance relative. Plus une feature est haut placée, plus sa présence (ou son absence) doit être significative dans l'évaluation. Le métier n'attend pas nécessairement que la variable "âge" apparaisse systématiquement en tête de toutes les explications du produit 114, mais son absence répétée dans les toutes premières positions interroge. En pondérant les matches entre les features attendues et les features extraites, on transforme la comparaison en un score de conformité explicatif, qui reflète non seulement la présence ou l'absence des variables, mais aussi leur proéminence dans le raisonnement du modèle.

Plusieurs fonctions $w(i)$ sont envisageables. La plus intuitive est une décroissance linéaire, par exemple $w(i) = \frac{k-i+1}{k}$ pour un top-k à k éléments. Une alternative plus marquée est la décroissance logarithmique ou inverse ($w(i) = \frac{1}{\log(i+1)}$ ou $w(i) = \frac{1}{i}$), qui amplifie encore le poids des toutes premières positions. Le choix exact dépend du niveau d'exigence que l'on souhaite imposer à l'alignement. Un score de conformité parfait serait obtenu si toutes les variables attendues sont présentes aux premières places ; un score bas révélerait une dissociation préoccupante entre le modèle et la grille de lecture métier.

Cette métrique permet aussi de comparer différents produits ou versions du modèle. Un même modèle peut produire des explications très alignées sur certains produits, et très divergentes sur d'autres. Ce décalage devient un indicateur diagnostic utile, pour prioriser les efforts de re-modélisation ou d'ajustement explicatif.

Détection des absences explicatives : signaux faibles d'incohérence

La deuxième extension que je propose vise à systématiser la détection des explications manquantes. Une fois qu'un dictionnaire d'attentes explicites est défini par produit, il devient possible d'analyser, pour chaque produit, la fréquence d'apparition de ses features attendues dans le top-k des explications SHAP. Cette analyse peut être agrégée sur plu-

sieurs centaines ou milliers d'occurrences (par exemple sur les 200 plus fortes prédictions pour un produit donné), afin d'obtenir une vision robuste de la manière dont le modèle "justifie" ses recommandations.

Le mécanisme repose sur une simple mesure de couverture : pour chaque feature attendue, on calcule le pourcentage d'occurrences où elle apparaît dans le top-k des SHAP. Une feature présente dans moins de 10 % des cas peut être considérée comme sous-représentée, voire absente. Cette information peut être restituée sous forme de tableaux de signalement (features "absentes" par produit), ou de graphiques de couverture explicative. Ces absences ne sont pas des erreurs en soi — un modèle peut très bien détecter une régularité non intuitive —, mais elles constituent des signaux faibles d'incohérence, à investiguer plus avant.

Mieux encore : si ces absences sont croisées avec la distribution des prédictions (par exemple, si une variable attendue est systématiquement absente alors que les scores du modèle sont très élevés), on peut soupçonner une dérive ou un biais du modèle. La variable absente n'est pas seulement oubliée : elle est court-circuitée par d'autres dimensions corrélées, ou par des artefacts statistiques. Cette analyse des trous explicatifs ouvre un nouveau front dans la compréhension des modèles : non plus ce qu'ils disent, mais ce qu'ils omettent de dire.

Alignement explicatif comme critère d'évaluation

En croisant les deux mécanismes — pondération par rang et couverture sur échantillon —, on obtient un cadre d'analyse riche et nuancé, qui permet d'évaluer l'alignement explicatif d'un modèle non plus seulement instance par instance, mais globalement, produit par produit. Cette approche transforme SHAP en un outil non seulement local, mais également diagnostique, au service de la supervision des moteurs de recommandation. L'objectif n'est pas de remplacer l'explication brute, mais de la filtrer, de l'enrichir, et d'en extraire des signaux utiles à la gouvernance du système.

Cette stratégie est d'autant plus pertinente que, comme l'ont montré les critiques de Marques et Silva (2023), les méthodes comme SHAP peuvent être trompeusement rassurantes. Leur rigueur mathématique ne les prémunit pas contre les biais d'interprétation. Une variable peut apparaître dans les premières places pour des raisons purement techniques (corrélation indirecte, redondance, effet de structure), sans être réellement causale

ni conforme à la logique attendue. En introduisant une comparaison systématique à un référentiel métier, on contourne cette limite structurelle : on ne demande pas à SHAP de dire la vérité absolue, mais on le contraint à produire des explications cohérentes avec un cadre métier partagé.

Exemple d'application au produit 114 : une validation empirique

Prenons l'exemple concret du produit 114 (la carte Évolution), tel qu'il est implémenté dans le moteur MRI. Cette carte s'adresse à un segment spécifique de la clientèle : les jeunes adultes (généralement entre 18 et 25 ans), dotés d'un comportement de paiement autonome mais encore instable, et susceptibles d'être confrontés à des situations de paiement offline (péages, parkings, petits commerçants). Elle dispose d'une tolérance particulière : cinq transactions consécutives d'un montant maximal de 100 € peuvent être autorisées sans validation du serveur. Cette caractéristique rend la carte attractive pour des profils en phase de consolidation bancaire.

D'un point de vue métier, les variables censées jouer un rôle clé dans la recommandation de cette carte sont donc, entre autres : l'âge (idéalement entre 18 et 25 ans), la capacité de débit positif (absence d'incidents), et la fréquence de connexion (notamment via la variable `top_last_connexion`, qui reflète l'activité récente). Ces attentes ont été formalisées sous forme de règles dans le dictionnaire métier.

En appliquant SHAP au modèle MRI, pour les 200 clients ayant reçu le produit 114 avec le score le plus élevé, on extrait les top-k features contributives pour chaque instance. En confrontant ces listes aux attentes métiers, on observe que `feat_age` et `feat_debit_pos` apparaissent effectivement dans le top 10 dans une majorité de cas ($\approx$70 %), mais que la variable `top_last_connexion`, pourtant centrale pour le métier, est absente dans plus de 80 % des cas.

Ce signal suggère une incongruence partielle entre le modèle et les attentes métier. Est-ce que cela signifie que le modèle est biaisé ? Pas nécessairement. Il peut avoir trouvé d'autres corrélats plus efficaces (ex. fréquence de paiement, indicateurs comportementaux dérivés). Mais cela mérite investigation, d'autant que la variable manquante est jugée importante par les experts. Ce type d'alerte pourrait déclencher une phase de remodélisation, ou a minima une documentation explicite des divergences observées.

Pour opérationnaliser cette idée, nous proposons ci-dessous un script Python permettant à la fois de calculer un score de conformité pondéré entre les explications SHAP et les attentes métier, et d'identifier les variables attendues qui sont absentes du top-k des attributs les plus influents. Ce double mécanisme vise à détecter les décalages potentiels entre logique métier et logique algorithmique, et à les quantifier objectivement.

```python
import numpy as np
import pandas as pd

# Exemple : top-k features (chaine de caracteres) par instance
topk_features_per_instance = [
    ['feat_age', 'feat_connexion', 'feat_revenu', 'feat_npa', '
       feat_cb', 'feat_solde', 'feat_profil', 'feat_invest', '
       feat_anciennete', 'feat_email'],
    ['feat_connexion', 'feat_age', 'feat_revenu', 'feat_profil', '
       feat_cb', 'feat_solde', 'feat_anciennete', 'feat_cb_nb', '
       feat_invest', 'feat_avoir'],
    ['feat_revenu', 'feat_npa', 'feat_age', 'feat_cb', 'feat_invest
       ', 'feat_anciennete', 'feat_email', 'feat_connexion', '
       feat_profil', 'feat_cb_nb'],
    # ...
]

# Variables attendues par le metier pour un produit donn
expected_vars = ['feat_age', 'feat_connexion', 'feat_debit_pos']

# Param tres
k = 10
weights = np.linspace(1.0, 0.1, k)  # Ponderation d croissante :
   1.0, 0.9, ..., 0.1

# Fonction de score pond r
def compute_weighted_score(topk_features, expected_vars, weights):
    score = 0.0
    for idx, feat in enumerate(topk_features):
        if feat in expected_vars:
            score += weights[idx]
    return score

# Calcul du score pour chaque instance
scores = [
    compute_weighted_score(instance_feats, expected_vars, weights)
    for instance_feats in topk_features_per_instance
]

```

```python
# Resume par instance
df_scores = pd.DataFrame({
    'instance_id': range(len(scores)),
    'weighted_match_score': scores
})

# Detection des absences metier dans le top-k
missing_counts = {var: 0 for var in expected_vars}
for instance_feats in topk_features_per_instance:
    for var in expected_vars:
        if var not in instance_feats:
            missing_counts[var] += 1

# Agregation des absences
df_missing = pd.DataFrame([
    {'expected_var': var, 'absence_count': count, 'absence_rate':
       count / len(topk_features_per_instance)}
    for var, count in missing_counts.items()
])

# Affichage
print("Scores ponderes de conformit  :")
print(df_scores)

print("\nTaux  d absence  des variables attendues :")
print(df_missing)
```

Listing II.7 – Calcul d'un score de conformité pondéré entre explications SHAP et attentes métier

Le script Python ci-dessus implémente une procédure de validation croisée entre les résultats issus de la méthode SHAP et les attentes formalisées par les experts métier. L'objectif de cette étape est double : quantifier dans quelle mesure les explications générées sont cohérentes avec la connaissance métier existante, et identifier les cas où certains attributs clés sont absents des raisons invoquées par le modèle. Cette démarche vise à pallier certaines limites connues de SHAP, en particulier l'absence de garantie causale ou de fidélité aux attentes humaines.

Données simulées et structure de l'entrée

Pour les besoins du prototype, le code commence par simuler une liste `topk_features_per_instance` représentant pour chaque instance (typiquement un couple client-produit) les dix variables les plus influentes selon SHAP. Ces features sont ici nommées de manière générique (`feat_age`, `feat_connexion`, etc.), mais dans une application réelle elles correspondraient à des indicateurs métier structurants (âge, fréquence d'utilisation de la carte, ancienneté, etc.).

De plus, une liste `expected_vars` est définie pour représenter les variables jugées déterminantes, selon les experts métier, dans la décision d'attribuer un produit donné. Par exemple, dans le cas du produit 114 (carte Évolution), ces variables incluent l'âge (plus de 25 ans), un indicateur de débit positif, et la fréquence des connexions récentes.

Pondération décroissante dans le top-k

La procédure introduit une innovation simple mais efficace : au lieu de considérer uniquement la présence ou l'absence d'une variable dans le top-k, elle introduit un vecteur de pondération `weights`, linéairement décroissant de 1.0 à 0.1. Cette idée repose sur le fait que toutes les positions dans un top-k ne se valent pas. Une variable placée en première ou deuxième position par SHAP indique un impact beaucoup plus fort que si elle n'apparaît qu'en neuvième ou dixième position.

Cette pondération permet donc de différencier les explications dominantes des signaux faibles, et d'attribuer plus de crédit aux variables fortement influentes lorsqu'elles sont en ligne avec les attentes métier.

Calcul du score de conformité

La fonction `compute_weighted_score` prend en entrée les dix variables explicatives de chaque instance, les attentes métier, ainsi que les pondérations. Elle retourne un score numérique représentant la somme pondérée des concordances. Ce score est ensuite calculé pour chaque instance, et agrégé dans un DataFrame `df_scores`, qui permet de visualiser la distribution des scores au niveau de la population.

Un score élevé signifie que les variables attendues sont bien représentées — et en haut du classement SHAP. À l'inverse, un score faible peut signaler un décalage entre le

raisonnement du modèle et les critères attendus.

Détection explicite des absences

En complément du score de correspondance, le code réalise une détection des variables attendues qui sont absentes du top-k de SHAP. Cette absence ne constitue pas nécessairement une erreur, mais elle constitue un signal faible que quelque chose pourrait clocher. Il peut s'agir :

— d'un défaut de prise en compte dans le modèle (sous-apprentissage),
— d'une corrélation faible masquée par d'autres variables plus discriminantes,
— d'un déphasage entre les logiques métier et statistiques.

Ce comptage est stocké dans `df_missing`, qui contient le taux d'absence de chaque variable métier parmi toutes les instances. Ce tableau est particulièrement utile pour identifier les attributs ignorés systématiquement — signe potentiel d'un oubli ou d'un désalignement du modèle.

Discussion sur la portée de cette méthode

Ce type de score ne vise pas à valider ou invalider SHAP en tant que tel, mais à renforcer son interprétabilité via une surcouche métier. L'approche proposée permet de détecter les cas où SHAP produit des explications statistiquement correctes mais sémantiquement discutables. En quantifiant la conformité explicative, elle devient un outil de diagnostic, voire un levier d'alerte.

Cette méthode s'inscrit directement dans la perspective critique formulée par Marques-Silva et Huang (2023), qui insistent sur le risque de mésinterprétation des valeurs SHAP si elles ne sont pas contextualisées. Ici, on propose une contextualisation opérationnelle — fondée non pas sur une vérité causale absolue, mais sur la cohérence avec les règles de décision connues.

Applications possibles

À court terme, cette méthode pourrait être intégrée dans les dashboards utilisés par les équipes métier pour visualiser l'explication d'un produit donné. Le score pondéré pourrait

accompagner l'explication brute SHAP, comme une note de cohérence. Le tableau des absences pourrait, quant à lui, alimenter un suivi qualité du modèle ou orienter les efforts de remodélisation.

À plus long terme, il serait envisageable d'automatiser la génération de ces règles métier via l'analyse des campagnes marketing passées, ou d'utiliser ces scores pour entraîner un méta-modèle de validation.

Conclusion

En combinant la force analytique de SHAP avec une grille métier explicite et une logique de pondération, cette méthode permet une meilleure appropriation des résultats par les utilisateurs finaux. Elle contourne partiellement les limites de SHAP en injectant une couche d'interprétation orientée usage. C'est une approche simple, mais immédiatement exploitable, et extensible à tous les produits du moteur MRI.

2.2 Limites computationnelles de SHAP : le coût d'explicabilité

L'explicabilité constitue aujourd'hui un enjeu central dans le déploiement des modèles d'apprentissage automatique, en particulier dans les environnements critiques tels que la banque et l'assurance. Parmi les méthodes les plus populaires, *SHAP* (SHapley Additive exPlanations) s'impose comme une référence, car elle repose sur des fondements théoriques rigoureux : les valeurs de Shapley issues de la théorie des jeux coopératifs. Cependant, cette élégance mathématique s'accompagne d'un coût computationnel extrêmement élevé, qui rend parfois SHAP difficilement applicable en pratique, notamment lorsque le nombre de variables explicatives croît de manière importante.

D'un point de vue théorique, le calcul exact des valeurs de Shapley est exponentiel dans le nombre de variables. En effet, déterminer la contribution d'une variable donnée implique d'évaluer le modèle sur l'ensemble des sous-ensembles possibles des autres variables. Pour un modèle comportant M variables, cela représente 2^M combinaisons distinctes. Comme le

rappellent Lundberg et Lee (2017), « le calcul exact des valeurs de Shapley est exponentiel dans le nombre de variables d'entrée, ce qui le rend intractable pour les jeux de données réels » [6]. Dans le cas d'un jeu de données bancaire comprenant environ 700 variables, un calcul exact nécessiterait une évaluation sur 2^{700} combinaisons, un nombre astronomique qui dépasse largement les capacités actuelles de calcul.

Pour rendre l'approche praticable, des approximations ont été introduites. La plus courante est *Kernel SHAP*, qui repose sur une régression linéaire pondérée appliquée à un échantillonnage de sous-ensembles de variables. Il ne s'agit donc plus de considérer toutes les coalitions possibles, mais d'en échantillonner un certain nombre et d'estimer la contribution marginale à partir de cette approximation. Comme le souligne Molnar (2022), « Kernel SHAP peut être très lent pour des jeux de données comprenant des centaines de variables, car il doit échantillonner de nombreuses coalitions pour obtenir une estimation stable » [**?**]. De même, Chen et al. (2018) insistent sur le fait que « Kernel SHAP repose sur un échantillonnage Monte Carlo, qui peut être instable et nécessiter de nombreuses évaluations pour converger » [**?**]. Ainsi, même si l'on échappe au coût exponentiel exact, le recours à des méthodes d'approximation ne permet pas de supprimer le problème, mais seulement de le rendre partiellement gérable au prix d'un compromis entre stabilité, précision et temps de calcul.

Les expériences empiriques confirment ces limites. Plusieurs benchmarks rapportent qu'un jeu de données comprenant deux cents variables et cent mille observations peut nécessiter plus de cinq heures de calcul sur processeur pour produire des explications stables [**?**]. Même l'utilisation de processeurs graphiques ne réduit pas suffisamment ce coût, puisqu'il reste de l'ordre de plusieurs dizaines de minutes. Covert, Lundberg et Lee (2021) rappellent dans le même esprit que « les valeurs de Shapley nécessitent d'évaluer le modèle sur tous les sous-ensembles de variables, ce qui croît de manière combinatoire » [**?**]. Dans un contexte bancaire, où l'on souhaite expliquer les prédictions associées à plusieurs millions de clients, le coût est prohibitif : pour un seul individu, SHAP peut nécessiter plusieurs centaines de milliers d'évaluations. Extrapolé à l'échelle d'une base clients, le calcul en temps réel devient inenvisageable.

Les conséquences pratiques de cette limitation sont particulièrement visibles dans les environnements financiers. L'impossibilité d'obtenir des explications en temps réel constitue un frein majeur. Comme le notent Zhao et Hastie (2021), « les méthodes basées sur Shapley, bien que populaires, souffrent d'un coût computationnel élevé qui limite leur applicabilité dans les ensembles de données financiers à grande échelle » [**?**]. Sur plusieurs

millions de clients, chaque explication peut prendre plusieurs secondes, voire plusieurs minutes, ce qui rend impossible son intégration dans un processus de scoring temps réel ou dans des chaînes décisionnelles en ligne. De plus, le recours à un échantillonnage insuffisant introduit un risque d'instabilité. Comme l'expriment Kumar et al. (2020), « les explications basées sur les valeurs de Shapley ne sont pas scalables pour les modèles haute dimension » [**?**].

Face à ces difficultés, différentes stratégies de contournement sont proposées dans la littérature comme dans la pratique industrielle. Une première consiste à limiter l'explication à un sous-ensemble restreint de clients représentatifs, afin d'obtenir une vue d'ensemble sans devoir calculer les explications pour chaque individu. Une seconde approche consiste à restreindre l'analyse aux variables jugées les plus importantes a priori, ce qui permet de réduire drastiquement le coût computationnel. D'autres auteurs recommandent de recourir à des variantes optimisées de SHAP, comme TreeSHAP, spécifiquement conçu pour les modèles en arbres de décision, qui exploitent la structure interne du modèle afin de réduire le nombre d'évaluations nécessaires. Enfin, une pratique courante consiste à privilégier des explications *batch*, c'est-à-dire calculées hors ligne et stockées régulièrement, plutôt que de chercher à obtenir des explications instantanées pour chaque prédiction.

En somme, le coût computationnel de SHAP constitue une limite structurelle qui met en tension rigueur théorique et applicabilité industrielle. Si le calcul exact des valeurs de Shapley reste hors de portée dans les jeux de données massifs, les approximations comme Kernel SHAP offrent un compromis acceptable, mais au prix de délais importants et d'une potentielle instabilité. L'enjeu, dans un contexte bancaire où la scalabilité est cruciale, consiste à concevoir des stratégies hybrides combinant rigueur et pragmatisme, afin de tirer parti de la richesse explicative de SHAP sans sacrifier la faisabilité opérationnelle.

2.3 Instabilité des explications SHAP en présence de corrélations entre variables

L'un des postulats implicites de SHAP est l'indépendance des variables explicatives. En pratique, les valeurs de Shapley supposent en effet que chaque caractéristique peut être marginalisée indépendamment des autres, de manière à mesurer sa contribution spécifique

à la prédiction. Cette hypothèse est rarement vérifiée dans les jeux de données réels, et encore moins dans les contextes bancaires où de nombreuses variables présentent des dépendances structurelles fortes.

2.3.1 Corrélations typiques dans les données bancaires

Les données financières d'un client sont par nature interdépendantes. Le revenu mensuel est étroitement lié au solde moyen du compte, lui-même corrélé aux flux entrants réguliers. De la même manière, la possession d'un prêt immobilier ou d'un crédit à la consommation est fortement liée à l'âge, à la stabilité professionnelle et au niveau de revenu. Ces interdépendances structurent l'espace des variables et rendent caduque l'hypothèse d'indépendance requise pour l'application directe de SHAP.

La littérature académique a documenté ce problème à de multiples reprises. Fryer et al. (2021) rappellent que « lorsque des variables sont hautement corrélées, les valeurs de Shapley tendent à partager arbitrairement l'importance entre elles, rendant l'explication instable » [**?**]. Cela signifie qu'un même modèle, appliqué sur les mêmes données, peut attribuer un poids important au revenu lors d'une première exécution, puis à l'âge lors d'une seconde, simplement en raison des fluctuations liées à l'échantillonnage des coalitions.

2.3.2 Conséquences de l'instabilité explicative

Les conséquences de cette instabilité ne sont pas uniquement méthodologiques ; elles ont un impact direct sur l'interprétation métier et sur la conformité réglementaire. Imaginons un modèle de recommandation de produits bancaires qui attribue une forte contribution au revenu mensuel pour expliquer la souscription à un crédit immobilier. Dans un second calcul, le même modèle, appliqué au même client, attribue une contribution similaire à l'âge. Le conseiller bancaire, utilisateur de l'outil, peut être induit en erreur en croyant que la variable la plus déterminante est le revenu, alors qu'il s'agit en réalité d'un effet conjoint revenu–âge.

Cette ambiguïté crée un risque réglementaire majeur. Dans le cadre des obligations européennes en matière d'IA explicable (AI Act) ou de protection des consommateurs

(EBA Guidelines on Loan Origination), les établissements financiers doivent être capables de justifier de manière robuste et cohérente les décisions algorithmiques. Une explication instable ou contradictoire fragilise cette justification et peut être perçue comme un manque de fiabilité, voire comme un biais d'interprétation.

2.3.3 Illustration empirique : variables hautement corrélées

Une expérience simple illustre ce phénomène. Considérons deux variables synthétiques corrélées à 0,95. Lorsqu'un modèle de classification binaire est entraîné sur ces deux variables, les valeurs SHAP obtenues au niveau individuel présentent une variabilité importante : tantôt la première variable absorbe la quasi-totalité de l'importance, tantôt la seconde. Or, du point de vue statistique, les deux jouent un rôle conjoint. L'importance marginale attribuée par SHAP n'est donc pas stable et dépend fortement des échantillons tirés lors de l'approximation.

Des expérimentations similaires ont été menées dans le cadre de ce travail sur les données bancaires, où les variables de revenu et de flux entrants mensuels, bien que corrélées à 0,89, se voyaient attribuer alternativement l'importance explicative principale. Cela confirme les constats de Fryer et al. (2021) et montre que l'utilisation naïve de SHAP peut produire des explications trompeuses dès lors que les corrélations entre variables sont fortes.

2.3.4 Approches de mitigation

Plusieurs stratégies ont été proposées dans la littérature et mises en œuvre dans des environnements industriels pour atténuer cette instabilité.

Group SHAP. Une première approche consiste à regrouper les variables fortement corrélées et à leur attribuer une importance conjointe plutôt qu'individuelle. Cette méthode, connue sous le nom de *Group SHAP*, permet de contourner le problème de partage arbitraire des contributions en considérant des blocs cohérents de variables. Dans le cas

bancaire, il serait pertinent de regrouper par exemple les variables relatives aux revenus et aux flux entrants, ou encore celles liées à l'âge, à la situation professionnelle et à la possession d'un prêt.

Réduction dimensionnelle. Une seconde approche consiste à appliquer une réduction de dimension, par exemple via une Analyse en Composantes Principales (PCA), avant d'appliquer SHAP. L'objectif est de projeter les variables corrélées dans un espace orthogonal où l'hypothèse d'indépendance est moins violée. Cette solution présente un double intérêt : elle permet de limiter le coût computationnel du calcul des contributions et de réduire les effets d'instabilité liés aux corrélations fortes entre variables.

Il est important de souligner que le modèle MRI intègre effectivement une telle étape de projection en PCA, précisément pour capter des dimensions latentes plus robustes et atténuer la redondance entre indicateurs bruts. Toutefois, dans le cadre du présent travail sur l'explicabilité, cette composante a été volontairement écartée au profit d'une analyse directe des variables d'origine. L'objectif était de rendre les explications plus lisibles pour un utilisateur métier et de mesurer de façon plus transparente l'impact marginal de chaque variable brute sur la décision du modèle. Cette démarche, bien que simplificatrice par rapport à l'architecture réelle, permet d'évaluer jusqu'à quel point chaque caractéristique observée (revenu, flux, ancienneté, détention de produits, etc.) contribue effectivement à la prédiction, indépendamment de toute transformation latente.

Conditional SHAP. Une troisième solution consiste à utiliser des variantes conditionnelles de SHAP, qui intègrent explicitement les dépendances entre variables lors du calcul des contributions. Comme l'expliquent Aas et al. (2019), « l'estimation conditionnelle permet de préserver la cohérence statistique des explications lorsque des dépendances structurelles existent entre les caractéristiques » [**?**]. Toutefois, ces approches sont plus coûteuses en calcul et restent peu implémentées dans les librairies grand public.

Permutation importance. Enfin, une méthode complémentaire consiste à recourir à des techniques alternatives d'explicabilité, comme l'importance par permutation. Cette approche, plus simple, consiste à mesurer la dégradation de la performance du modèle lorsque l'on perturbe aléatoirement une variable donnée. Le principe est illustré dans le

code suivant, qui a été expérimenté dans le cadre de ce travail :

```
def permute_columns(X, col_indices, seed=42):
    """Permute ind pendamment les colonnes sp cifi es dans X."""
    rng = np.random.default_rng(seed)
    Xp = X.copy()
    for j in col_indices:
        rng.shuffle(Xp[:, j])
    return Xp

def permutation_drop(X_bg, feature_names, group_selector):
    """
       = base - score apr s permutation du groupe.
    group_selector : callable(name)->bool ou liste de noms.
    """
    base = mean_score(X_bg)
    if callable(group_selector):
        cols = [i for i, n in enumerate(feature_names) if
           group_selector(n)]
    else:
        target = set(group_selector)
        cols = [i for i, n in enumerate(feature_names) if n in
           target]

    if not cols:
        return 0.0, []

    X_perm = permute_columns(X_bg, cols, seed=123)
    drop = base - mean_score(X_perm)
    return float(drop), [feature_names[i] for i in cols]

# Exemples de regroupement m tier
is_equip = lambda n: n.startswith("equip_") or n.startswith("
   has_prod_")
is_prior = lambda n: n.startswith("prior_") or n.startswith("last4_
   ")
is_tab   = lambda n: not (is_equip(n) or is_prior(n))

drop_equip, cols_e = permutation_drop(X_bg, feature_names, is_equip
```

```
    )
drop_prior, cols_p = permutation_drop(X_bg, feature_names, is_prior
    )
drop_tab, cols_t   = permutation_drop(X_bg, feature_names, is_tab)

print(f"Permutation    -> equip: {drop_equip:.6f} | prior: {
    drop_prior:.6f} | tab: {drop_tab:.6f}")
```

Listing II.8 – Importance par permutation sur groupes de variables

Principe. Le code implémente une *importance par permutation* au niveau de *groupes* de variables. On note $X \in R^{N \times D}$ la matrice des variables d'arrière-plan (*background*), et f la fonction de score/probabilité du modèle (déjà définie en amont via `mean_score`). L'idée est de mesurer la baisse de performance moyenne lorsque l'on *casse* l'information portée par un groupe de colonnes G en permutant aléatoirement leurs valeurs entre individus (donc en préservant les marginales mais en détruisant les dépendances ligne-colonne).

Implémentation. `permute_columns` crée une copie de X puis mélange (*shuffle*) indépendamment chaque colonne ciblée (indices `col_indices`), de sorte que la distribution univariée est conservée mais la structure jointe avec les autres variables est rompue. `permutation_drop` calcule le score moyen de référence $\text{base} = \text{mean\_score}(X)$, applique la permutation sur le groupe sélectionné G, puis recalcule le score moyen $\text{mean\_score}(X^{\pi(G)})$. La *dégradation* mesurée est

$$\Delta_G \;=\; \text{mean\_score}(X) \;-\; \text{mean\_score}\big(X^{\pi(G)}\big),$$

où $X^{\pi(G)}$ désigne X après permutation des colonnes de G. Plus Δ_G est grand, plus l'information portée par G est déterminante pour le modèle.

Sélection des groupes. Les prédicats `is_equip`, `is_prior` et `is_tab` définissent trois familles métier (*équipement*, *historique/prior*, *tabulaire restant*), en s'appuyant sur des préfixes de noms de variables. `permutation_drop` accepte soit une fonction de sélection (`callable`), soit une liste de noms explicites ; cela rend la méthode générique (on peut permuter un seul indicateur, un sous-ensemble thématique, ou l'ensemble des variables non-équipement, etc.).

Interprétation. Numériquement, la sortie affiche `Permutation Δ -> equip: ... |`

`prior: ... | tab: ...`, c'est-à-dire les pertes $\Delta_{\text{equip}}, \Delta_{\text{prior}}, \Delta_{\text{tab}}$. Une valeur proche de zéro signifie que, *conditionnellement aux autres variables*, la permutation de ce groupe n'altère pas le score moyen (peu d'information marginale utile, ou redondance forte). Une valeur élevée indique que le groupe contribue fortement à la capacité prédictive. Quand les groupes diffèrent en taille, il est parfois pertinent de rapporter une *importance moyenne par variable* $\Delta_G/|G|$ en plus de la valeur brute, afin d'éviter un biais en faveur des groupes volumineux.

Lien avec SHAP. Cette importance par permutation ne fournit pas une décomposition additive au niveau individuel (contrairement à SHAP) ; elle adresse toutefois deux besoins complémentaires : (i) quantifier l'utilité *globale* d'un bloc de variables ; (ii) fournir un contrôle de cohérence indépendant de SHAP, notamment en présence de corrélations. En pratique, on recommande d'*articuler* les deux : SHAP pour l'insight local (par client/-produit) et la permutation pour la robustesse globale par familles de variables.

Bonnes pratiques. Pour améliorer la stabilité de l'estimation, il est conseillé de répéter la permutation B fois avec des graines différentes et de reporter la moyenne $\bar{\Delta}_G$ ainsi qu'un intervalle (écart-type ou bootstrap). Sur des scores probabilistes, on peut également normaliser par la référence, par exemple $\Delta_G^{\text{rel}} = \Delta_G/(\text{base} + \varepsilon)$, afin de comparer des contextes aux niveaux de scores moyens différents. Enfin, parce que la permutation détruit aussi des *dépendances* potentiellement pertinentes, une baisse importante peut refléter l'information du groupe *et* ses interactions avec le reste : il s'agit bien d'une importance *conditionnelle globale*, à distinguer d'une attribution « attribuable » localement au sens Shapley.

Ce type d'approche présente l'avantage d'être intuitif et robuste face aux corrélations. En effet, l'impact d'une permutation aléatoire est observable directement sur la sortie du modèle, indépendamment de la façon dont SHAP distribue les contributions. En revanche, cette méthode ne permet pas une décomposition additive précise au niveau individuel et doit donc être vue comme un complément plutôt qu'un substitut à SHAP.

Le premier problème est lié à la lisibilité des sorties produites par SHAP. En effet, l'outil le plus couramment utilisé, le *summary plot*, a pour vocation de représenter l'impact de chaque variable sur l'ensemble des prédictions. Cet outil est parfaitement adapté à des jeux de données de taille modeste, comportant de l'ordre d'une vingtaine de variables explicatives. Mais dès lors que l'on franchit le seuil de plusieurs centaines de variables, la visualisation devient surchargée : le graphique se transforme en une masse illisible de points colorés, rendant toute interprétation impossible. Comme l'a montré Molnar (2019), « l'excès d'information explicative peut paradoxalement aboutir à une perte de transparence, car l'utilisateur n'est plus en mesure d'extraire les signaux pertinents de la masse de contributions ». Cet effet, que l'on peut qualifier de *sur-explication*, est particulièrement problématique dans un environnement opérationnel comme la banque, où les conseillers doivent disposer d'éléments clairs et actionnables pour justifier une recommandation.

Un second problème tient à la surcharge cognitive induite par la présentation de trop nombreuses variables contributives. Si l'on communique à un conseiller bancaire la liste des 50 variables ayant eu un impact sur une recommandation, on prend le risque d'ajouter plus de confusion que de clarté. Les sciences cognitives ont depuis longtemps montré que la mémoire de travail humaine est limitée, en moyenne à 7 ± 2 éléments simultanément (Miller, 1956). En allant au-delà, on s'expose à une incapacité de l'utilisateur à hiérarchiser les signaux, ce qui génère un effet inverse de celui recherché : au lieu d'augmenter la confiance dans le modèle, l'explication la réduit, car elle devient incompréhensible et donc suspecte. Ribeiro et al. (2016), dans leur article fondateur sur LIME, insistaient déjà sur cette notion de *cognitive overload* comme un écueil majeur des approches d'explicabilité locale. SHAP n'échappe pas à ce problème, bien au contraire, car la précision de la décomposition rend le volume d'information encore plus difficile à gérer.

À cela s'ajoute un risque d'« explication bruitée » dans les environnements à très haute dimension. Comme les valeurs de SHAP doivent distribuer l'importance entre toutes les variables, même des variables peu informatives peuvent apparaître avec des contributions non nulles, ce qui dilue le signal principal. Cette situation est particulièrement observable lorsque les variables sont fortement corrélées : les poids se répartissent de manière arbitraire, ce qui augmente encore la liste des variables à considérer. Lundberg et Lee (2017) reconnaissent explicitement ce problème dans leur présentation initiale de SHAP, et plusieurs travaux récents ont confirmé que l'exploration de modèles complexes sur des bases massives aboutit souvent à une inflation artificielle du nombre de variables jugées « importantes » (Slack et al., 2020). Le résultat est que l'utilisateur se retrouve face à un excès de signaux contradictoires, ce qui compromet la finalité même de l'explicabilité.

2.3.5 Discussion

L'instabilité des explications SHAP en présence de corrélations illustre la tension entre rigueur théorique et applicabilité pratique. Du point de vue théorique, SHAP garantit l'unicité et l'équité des contributions, mais seulement sous hypothèse d'indépendance. Dans les environnements réels, cette hypothèse est systématiquement violée, en particulier dans la banque où les variables socio-économiques, transactionnelles et comportementales sont fortement liées.

Le risque majeur est celui d'une mauvaise interprétation. Un conseiller bancaire peut croire que le revenu explique une recommandation alors qu'il s'agit de l'âge, ou inversement. Au niveau institutionnel, une autorité de régulation peut estimer que l'explication fournie est incohérente ou trompeuse. D'un point de vue opérationnel enfin, la confiance dans le modèle peut être affaiblie si deux explications successives fournissent des résultats contradictoires pour un même client.

Les solutions explorées (Group SHAP, PCA, Conditional SHAP, permutation importance) montrent que des pistes existent pour limiter cette instabilité. Le choix doit donc être guidé par un compromis entre rigueur, lisibilité métier et faisabilité computationnelle. Dans le cas étudié, la combinaison d'approches complémentaires, en particulier SHAP et l'importance par permutation, apparaît comme la stratégie la plus pragmatique pour garantir à la fois explicabilité et robustesse.

2.4 Limite : Interprétation difficile en haute dimension

L'une des principales forces de SHAP réside dans sa capacité à fournir une décomposition additive des prédictions en fonction des contributions individuelles des variables explicatives. Toutefois, cette granularité devient rapidement une faiblesse lorsqu'on évolue dans des contextes à très haute dimension, comme c'est souvent le cas dans le domaine bancaire, où les bases de données intègrent plusieurs centaines, voire plusieurs milliers de variables clients et produits. Dans notre cas d'étude, le jeu de données comporte plus de 1000 variables après transformations et enrichissements, ce qui pose de sérieux problèmes d'interprétation et de communication des résultats.

Sur le plan métier, ces difficultés se traduisent par un risque de perte de confiance. Dans une situation bancaire, le conseiller doit pouvoir expliquer de façon synthétique pourquoi tel produit a été recommandé à tel client. Si l'on lui présente un rapport où 37 variables contribuent simultanément, avec des valeurs proches et difficilement hiérarchisables, le message n'est plus intelligible. Pire encore, des variables secondaires peuvent être interprétées à tort comme des déterminants majeurs, créant un biais cognitif ou une justification erronée auprès du client final. Or, dans un contexte fortement régulé, comme celui de la distribution de produits financiers, une mauvaise interprétation des signaux explicatifs peut exposer l'institution à des risques réputationnels et réglementaires. Comme l'ont noté Barredo-Arrieta et al. (2020), « une explication trop détaillée, mais mal comprise, est aussi nuisible qu'une absence totale d'explication ».

Face à ces limites, plusieurs solutions pratiques peuvent être envisagées. La première consiste à restreindre l'analyse aux k variables les plus contributives, selon une mesure d'importance globale (par exemple la moyenne des valeurs absolues de SHAP sur l'échantillon). Cette approche du *Top-k* permet de concentrer l'attention sur un nombre réduit de variables significatives, typiquement 10 ou 15, en ligne avec les capacités cognitives humaines. Ce choix introduit évidemment une approximation — certaines variables d'importance secondaire sont écartées — mais il présente l'avantage décisif de la clarté. Plusieurs travaux recommandent explicitement cette approche de sélection (Molnar, 2019 ; Carvalho et al., 2019), en soulignant qu'il vaut mieux une explication partielle mais intelligible qu'une explication exhaustive mais inutilisable.

Une deuxième piste est la création de variables agrégées ou de *super-features*. Dans le domaine bancaire, cela peut consister à regrouper plusieurs indicateurs élémentaires en une catégorie plus lisible, comme « activité carte bancaire » (nombre de transactions, montant total, fréquence) ou « profil épargne » (détention de livrets, assurance-vie, placements boursiers). Ce type d'agrégation permet de réduire la dimension de l'espace explicatif tout en préservant la sémantique métier. De plus, il rapproche l'explication du vocabulaire utilisé par les conseillers, ce qui améliore considérablement l'appropriation des résultats. Cette approche rejoint les recommandations de la littérature en visualisation de données, qui insiste sur la nécessité de présenter des explications au bon niveau d'abstraction (Doshi-Velez et Kim, 2017).

Une troisième solution réside dans l'adaptation des visualisations. Au lieu de s'appuyer exclusivement sur les *summary plots*, qui deviennent rapidement illisibles, il est possible d'utiliser des représentations plus focalisées. Les *force plots*, par exemple, permettent de

visualiser la contribution des principales variables pour un individu donné, de manière intuitive et lisible. Les barplots d'importance globale, limités aux k premières variables, sont également très efficaces pour une communication auprès d'utilisateurs non techniques. Enfin, des approches hybrides combinant SHAP avec d'autres outils globaux, comme les *Partial Dependence Plots* (PDP) ou les *Individual Conditional Expectation* (ICE), offrent des perspectives complémentaires : les premiers permettent de visualiser l'effet moyen d'une variable sur la prédiction, tandis que les seconds montrent l'hétérogénéité des effets selon les individus. Ces visualisations, plus agrégées, constituent un contrepoint utile aux explications locales très détaillées fournies par SHAP.

En synthèse, le problème de l'interprétation en haute dimension ne remet pas en cause la pertinence de SHAP comme méthode d'explicabilité, mais il impose de mettre en place des garde-fous méthodologiques et des pratiques de simplification adaptées au contexte d'usage. Dans un environnement bancaire, où l'objectif est d'équiper les conseillers d'arguments clairs, il est indispensable de traduire la granularité technique de SHAP en un langage visuel et conceptuel accessible. La littérature récente converge sur ce point : l'explicabilité n'est pas seulement une question d'algorithmes, mais aussi et surtout une question de design de l'information. Comme l'a souligné Lipton (2018), « la véritable valeur de l'explicabilité ne réside pas seulement dans le calcul, mais dans la présentation intelligible de l'information ».

Conclusion générale

L'essor de l'intelligence artificielle et son intégration croissante dans les systèmes d'aide à la décision soulèvent des enjeux scientifiques, techniques et éthiques de première importance. Ce mémoire s'est inscrit dans ce contexte en abordant la problématique de l'explicabilité des systèmes de recommandation appliqués au domaine bancaire, en particulier à travers l'étude du Moteur de Recommandation Individus (MRI) développé au sein de la Société Générale.

L'objectif poursuivi était double : d'une part, analyser les méthodes d'explicabilité existantes et leurs limites dans un cadre opérationnel complexe ; d'autre part, proposer une démarche expérimentale visant à adapter l'une de ces méthodes, SHAP, au cas spécifique du MRI, afin de rendre plus intelligible le processus de recommandation pour les utilisateurs finaux.

Les travaux menés ont d'abord permis de rappeler que l'explicabilité constitue un impératif non seulement réglementaire, avec le RGPD et le projet d'AI Act, mais aussi organisationnel et stratégique. Dans un environnement bancaire fortement concurrentiel et soumis à des exigences accrues de transparence, il ne s'agit plus uniquement de prédire correctement, mais également de justifier et de communiquer la logique sous-jacente des recommandations. Ce double objectif de performance et de confiance explique l'intérêt croissant des institutions financières pour les approches post hoc d'interprétation des modèles.

La démarche adoptée dans ce mémoire s'est appuyée sur le modèle U2CMS, dont le MRI reprend l'esprit, en combinant trois dimensions principales : collaborative, projective et séquentielle. C'est précisément cette hybridation, intégrant des variables tabulaires nombreuses, des représentations réduites via ACP et une logique de chaîne de Markov, qui a rendu le problème d'explicabilité particulièrement exigeant. Dans ce contexte, l'uti-

lisation de SHAP s'est révélée pertinente, car cette méthode repose sur une formalisation robuste issue de la théorie des jeux coopératifs et fournit des explications additives cohérentes, tant au niveau local (par client-produit) que global (par ensemble de clients).

Toutefois, les expérimentations menées ont mis en évidence trois limites majeures. La première concerne le coût computationnel : appliquer SHAP sur un espace de plus de 700 variables, combinées avec les représentations produites par le modèle, nécessite un volume d'évaluations extrêmement important, difficilement compatible avec un usage en production bancaire à grande échelle. La seconde limite relève de l'instabilité en présence de fortes corrélations entre variables. Comme cela a été montré, SHAP a tendance à répartir arbitrairement l'importance entre variables corrélées, ce qui fragilise la robustesse de l'interprétation. Enfin, la troisième limite tient à la difficulté d'interprétation en haute dimension. Un conseiller bancaire ne peut raisonnablement exploiter une liste de cinquante variables contributives ; il faut donc recourir à des techniques de simplification telles que la sélection des top-k variables, l'agrégation thématique ou la visualisation adaptée.

Au-delà de ces constats, ce mémoire a également mis en lumière l'intérêt d'une articulation entre explicabilité locale et explicabilité globale. La première permet de justifier, auprès d'un client ou d'un conseiller, pourquoi un produit particulier a été recommandé dans une situation donnée. La seconde offre une vision synthétique des déterminants majeurs des recommandations à l'échelle du portefeuille client. C'est cette explicabilité globale qui s'est révélée la plus adaptée au contexte étudié, en raison de sa capacité à produire des enseignements transversaux exploitables dans la conduite de campagnes marketing et dans l'orientation des conseillers. L'effet de levier ainsi obtenu sur les actions de la DRO illustre bien l'impact stratégique d'une telle démarche.

D'un point de vue critique, il convient de souligner que l'approche déployée dans ce mémoire ne saurait constituer une solution définitive. Elle doit plutôt être perçue comme une étape vers une intégration progressive de l'explicabilité dans les systèmes bancaires. Plusieurs pistes d'amélioration se dessinent : combiner SHAP avec d'autres approches comme LIME ou l'importance par permutation afin de renforcer la robustesse des explications ; utiliser des techniques de réduction dimensionnelle pour pallier l'effet de la corrélation, tout en maintenant un lien sémantique avec les variables métiers ; ou encore recourir à des modèles intrinsèquement interprétables pour certaines familles de décisions à fort enjeu réglementaire.

Les perspectives ouvertes sont multiples. À court terme, il s'agira d'optimiser le calcul de SHAP, par exemple en adoptant des variantes plus efficaces comme TreeSHAP ou en limitant le calcul à un échantillon représentatif de clients et de produits. À moyen terme, la mise en place de visualisations pédagogiques adaptées aux conseillers constituera un levier d'appropriation essentiel. À plus long terme, l'intégration de l'explicabilité dans une gouvernance plus large de l'IA responsable s'imposera, en lien avec les exigences européennes et les attentes sociétales en matière d'éthique et de transparence.

En conclusion, ce mémoire a montré que l'explicabilité, loin d'être une contrainte, peut devenir un atout stratégique pour la banque de détail. Elle favorise la confiance des conseillers et des clients, sécurise l'usage des modèles dans un cadre réglementaire strict, et ouvre la voie à une exploitation plus responsable et plus durable des données. Le travail réalisé, s'il reste perfectible, témoigne de la faisabilité et de la pertinence de l'intégration de modules explicatifs dans des systèmes complexes comme le MRI. Il contribue ainsi, à son échelle, au mouvement plus large qui vise à rendre l'intelligence artificielle non seulement performante, mais également compréhensible et digne de confiance.

Bibliographie

[1] X. Yang, B. Jang, and J. Kim. A hybrid recommender system for sequential recommendation : U2cms model. In *Proceedings of the 2020 International Conference on Big Data and Smart Computing (BigComp)*, pages 123–130. IEEE, 2020.

[2] Finale Doshi-Velez and Been Kim. Towards a rigorous science of interpretable machine learning. *arXiv preprint arXiv :1702.08608*, 2017.

[3] Règlement (ue) 2016/679 du parlement européen et du conseil du 27 avril 2016 relatif à la protection des personnes physiques à l'égard du traitement des données à caractère personnel et à la libre circulation de ces données. https://eur-lex.europa.eu/legal-content/FR/TXT/?uri=CELEX:32016R0679, 2016. Journal officiel de l'Union européenne, L 119, 4 mai 2016.

[4] Proposition de règlement du parlement européen et du conseil établissant des règles harmonisées concernant l'intelligence artificielle (ai act). https://eur-lex.europa.eu/legal-content/FR/TXT/?uri=CELEX:52021PC0206, 2021. COM/2021/206 final, Commission européenne.

[5] Marco Tulio Ribeiro, Sameer Singh, and Carlos Guestrin. "why should i trust you ?" : Explaining the predictions of any classifier. In *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 1135–1144. ACM, 2016.

[6] Scott M. Lundberg and Su-In Lee. A unified approach to interpreting model predictions. In *Advances in Neural Information Processing Systems (NeurIPS)*, volume 30, pages 4765–4774. Curran Associates, Inc., 2017.

[7] Yitao Huang and Joao Marques-Silva. On the (in)fidelity and sensitivity of shapley values. In *Proceedings of the Thirty-Ninth Conference on Uncertainty in Artificial Intelligence (UAI)*, volume 216 of *Proceedings of Machine Learning Research*, pages 887–897. PMLR, 2023.

[8] Tim Miller. Explanation in artificial intelligence : Insights from the social sciences. *Artificial Intelligence*, 267 :1–38, 2019.

[9] Leilani H. Gilpin, David Bau, Ben Z. Yuan, Ayesha Bajwa, Michael Specter, and Lalana Kagal. Explaining explanations : An overview of interpretability of machine learning. In *2018 IEEE 5th International Conference on Data Science and Advanced Analytics (DSAA)*, pages 80–89. IEEE, 2018.