RÉPUBLIQUE DU BÉNIN
MINISTÈRE DE L'ENSEIGNEMENT SUPÉRIEUR
ET DE LA RECHERCHE SCIENTIFIQUE

UNIVERSITÉ D'ABOMEY-CALAVI

INSTITUT DE FORMATION ET DE RECHERCHE EN INFORMATIQUE

BP 526 Cotonou Tel : +229 21 14 19 88
http://www.ifri-uac.com Courriel : contact@ifri.uac.bj

MÉMOIRE

pour l'obtention du

Diplôme de Licence en Informatique

Option : Genie Logiciel

Présenté par :
Ogbinto Samir Tafel **BONI**

Modélisation du risque de crédit avec de l'apprentissage automatique : cas de la Mutuelle pour le Développement à la Base (MDB)

Sous la supervision :
Dr Ing. Vinasétan Ratheil **HOUNDJI**

Membres du jury :

Année Académique : **2020-2021**

Sommaire

Dédicace

A

Ma famille et en particulier ma mère **TOUKEM Emmanuelle Lucie** pour son soutient sans faille.

Remerciements

Je tiens à remercier :

- Le tout puissant ALLAH pour m'avoir permis d'achever ce travail ;
- le Professeur Eugène C. Ezin, Directeur de l'Institut de Formation et de Recherche en Informatique (IFRI) ;
- Dr Ing Ratheil V. Houndji, encadreur de ce mémoire, pour sa disponibilité, sa patience et ses précieux conseils ;
- Monsieur LOUKPEY Boris, notre maître de stage pour sa pédagogie, sa disponibilité et pour avoir cru en moi ;
- Monsieur Dèdji Brian Whannou pour ses précieux conseils ;
- Le personnel de la MDB ;
- L'administration de l'IFRI ;
- Mes professeurs de l'IFRI ;
- Mes camarades et amis.

Résumé

L'évaluation du risque de crédit est une activité centrale pour tout établissement de crédit. L'approche traditionnelle utilisée par la plupart des Institutions de Microfinance (IMF) pour évaluer ce risque s'accentue autour d'une analyse technique et financière des données récoltées chez le client ainsi que sur le jugement et le bon sens. Cette approche ne permet pas toujours de déceler les bons des mauvais clients, il urge alors de développer des systèmes informatiques automatisés en complément aux méthodes existantes. C'est dans cette optique que nous mettons en place un système de modélisation du risque de crédit en complément aux méthodes existantes, capable de prédire la probabilité de défaut de remboursement d'un client emprunteur. Pour y arriver, nous utilisons de l'apprentissage automatique pour apprendre à déceler les bons des mauvais clients et ainsi prédire la probabilité de défaut de remboursement d'un client, pour un prêt individuel. Pour notre travail nous utilisons un ensemble de données de 7240 lignes de données, issu de la combinaison de plusieurs tables contenant des observations sur certains prêts antérieurs effectués à la Mutuelle pour le Dévéloppement à la Base (MDB). Nous avons procédé au nettoyage, à l'analyse exploratoire et au prétraitement des données. Une fois les étapes précédentes effectués, nous avons construit plusieurs modèles de machine Learning à savoir : Un modèle de logistic regression (regression logistique), un modèle de random forest classifier (forêts d'arbres décisionnels) et un modèle de Gradient Boosting Machine (amplification de gradient). Nous avons ensuite, procédé à la sélection du meilleur modèle et à une sélection des paramètres optimaux de ce dernier sur la base de plusieurs procédés, le meilleurs modèle était celui du Gradient Boosting Machine avec une précision de 0.89 et un rappel de 0.60. Nous avons après cela procédé à la validation et à la vérification des performances de notre modèle de Gradient Boosting Machine. Enfin une API (Application Programming Interface) permettant l'usage de ce système a été construit. Nous avons montré, à travers nos résultats, que notre solution peut s'inscrire dans un schéma de complémentarité avec les méthodes d'analyse du risque de crédit, existantes au sein de la MDB.

Mots clés : IMF, MDB, risque de crédit, machine learning

Abstract

Credit risk assessment is a central activity for any credit institution. The traditional approach used by most Microfinance Institutions (MFIs) to assess this risk is based on a technical and financial analysis of the data collected from the client as well as on judgment and common sense. This approach does not always make it possible to detect the good ones from the bad customers, it is therefore urgent to develop automated computer systems in addition to the existing methods. It is with this in mind that we are setting up a credit risk modeling system in addition to existing methods, capable of predicting the probability of default by a borrowing client. To do this, we use machine learning to learn how to spot good customers from bad customers and thus predict the likelihood of a customer defaulting on an individual loan. For our work we use a data set of 7240 lines of data, resulting from the combination of several tables containing observations on certain previous loans made to the Mutuelle pour le Développement à la Base (MDB). We performed data cleaning, exploratory analysis and pre-processing. Once the previous steps have been completed, we have built several machine learning models, namely: A logistic regression model (logistic regression), a random forest classifier model (decision tree forests) and a Gradient Boosting Machine model (amplification gradient). We then proceeded to the selection of the best model and a selection of the optimal parameters of the latter on the basis of several processes, the best model was that of the Gradient Boosting Machine with a precision of 0.89 and a recall of 0.60. We then proceeded to validate and verify the performance of our Gradient Boosting Machine model. Finally, an API (Application Programming Interface) allowing the use of this system has been built. We have shown, through our results, that our solution can be part of a scheme of complementarity with the credit risk analysis methods, existing within the MDB.

Key words: MFI, MDB, credit risk, machine learning

Liste des figures

Liste des tableaux

Sigles et abréviations

AA : Apprentissage Automatique 7

ANOVA : **An**alysis **of Va**riance 22, *Glossaire :* Analyse de la variance

API : Application Programming Interface iv, 26, *Glossaire :* Interface de programmation d'application

AUROC : Area Under the Receiver Operating Characteristics 10, *Glossaire :* Zone sous la courbe ROC

IA : Intelligence Artificielle 7, 8, *Glossaire :* Intelligence Artificielle

IV : Information Value 10, *Glossaire :* Valeur de l'information

WoE : Weight of Evidence 10, *Glossaire :* Poids de la preuve

Glossaire

Algorithme d'amplification de gradient :

c'est un groupe d'algorithmes d'apprentissage automatique qui combinent de nombreux modèles d'apprentissage faibles pour créer un modèle prédictif solide. 11

Analyse de la variance :

C'est une méthode statistique utilisée pour vérifier si les moyennes de trois groupes ou plus sont différentes. ANOVA utilise des tests F pour tester statistiquement l'égalité des moyennes (La statistique F est un ratio de deux écart-types). viii

Coefficient de Gini :

L'indice (ou coefficient) de Gini est un indicateur synthétique permettant de rendre compte du niveau d'inégalité pour une variable et sur une population donnée 10

Cross-Validation :

La validation croisée (« cross-validation ») est, en apprentissage automatique, une méthode d'estimation de fiabilité d'un modèle fondé sur une technique d'échantillonnage. 26

GridSearchCV :

c'est une technique qui utilise une combinaison différente de tous les hyperparamètres spécifiés et de leurs valeurs, calcule les performances de chaque combinaison et sélectionne la meilleure valeur pour les hyperparamètres. 11, 26

Hyperparamètres :

Les hyperparamètres sont des paramètres réglables qui vous permettent de contrôler le processus d'entraînement du modèle. 26

Intelligence Artificielle :

L'Intelligence Artificielle est la simulation de processus d'intelligence humaine par des machines. Plus particulièrement, des systèmes informatiques viii, 6

Interface de programmation d'application :

C'est un ensemble de routines, de protocoles, et d'outils utilisés pour concevoir des applications. Le but d'une API est de rendre facile la création d'un programme informatique en donnant aux développeurs l'accès à des blocs de code pré-faits et modifiables à souhait. viii

Poids de la preuve :

WoE est une mesure du pouvoir prédictif d'une variable indépendante par rapport à la variable cible, il mesure dans quelle mesure une caractéristique spécifique peut différencier les classes cibles. viii

sur-ajustement :

On parle de sur-ajustement lorsqu'on constate que le modèle offre de bons résultats sur les données de formation mais des résultats médiocres sur les données d'évaluation. 26

Valeur de l'information :

IV est une messure qui aide à classer nos fonctionnalités en fonction de leur importance relative viii

Zone sous la courbe ROC :

C'est une mesure de performance utiliser pour évaluer les modèles de classification. AUROC indique si un modèle est capable de classer correctement les classes, Il est également écrit comme AUC-ROC. viii

Introduction Générale

Le phénomène de digitalisation désormais observé dans pratiquement tout les domaines socio-économiques contribut à la génération d'une grande quantité de données. L'analyse de ces données par des humains peut s'avérer être une tâche difficile. Ainsi, plusieurs disciplines dont l'apprentissage automatique s'y attellent afin d'extraire un savoir ou de faire ressortir des structures intéressantes à partir de ces données dans le but de résoudre des problèmes ou d'améliorer des solutions existantes. Quand l'on s'intéresse au secteur de la microfinance et plus particulièrement au service d'octroie de crédit fournit par ces derniers, une grande quantité d'informations contenant des tendances et des modèles cachés est également générée, de part toutes les interactions effectuées et données produites par les acteurs de ce domaine. La présente étude s'articulera autour de l'application de techniques en apprentissage automatique pour améliorer la prise de décision et donc rehausser le taux de succès dans ce domaine.

Prolématique

La MDB est une des institutions mutualistes et l'une des plus connue de microfinance au Bénin. Dans le cadre de sa fonction d'intermédiation financière, la MDB s'expose au risque de ne pas recouvrer la totalité des fonds engagés dans les délais impartis et les créances douteuses que connait cette dernière ne sont pas sans effet sur ses résultats financiers. Par ailleurs, le marché du crédit de la microfinance mettant en relation le préteur et le client emprunteur est caractérisé par les asymétries d'information et les coûts de transactions plus élevés que celles des autres établissements financiers. D'après le portail FinDev dans sa publication "Taux d'intérêt en microfinance" fait en 2013, les IMF effectuent toutes leurs transactions en argent liquide et doivent souvent se déplacer pour collecter l'argent, ce qui occasionne des coûts opérationnels élevés (personnel, véhicules, agences, etc.), coûts que les banques traditionnelles n'ont pas à supporter. De manière générale, le processus d'analyse et d'octroi des prêts individuels se révèle relativement plus coûteux (temps, ressources humaines et financières) pour les IMF, en comparaison aux prêts bancaires. Cela implique une augmentation des coûts de transaction pour les SFD et dans un contexte d'asymétrie d'information, il devient plus difficile et compliqué de sélectionner les « bons » clients et de les différencier des « mauvais ». Pour octroyer le crédit à ses clients, la MDB procède à l'analyse technique et financière des données récoltées chez le client, ce qui ne permet pas à cette dernière d'évaluer avec une grande précision le risque d'impayé sur le crédit. Pour une meilleure analyse des informations récoltées sur les clients il faut trouver une autre approche complémentaire permettant de limiter au maximum le risque d'impayé. Ceci pourrait

passer par le recours à des systèmes informatiques d'évaluation du risque d'impayé sur les crédits octroyés par la MDB. Dans ce travail nous allons mettre en place un outil de modélisation du risque crédit capable de déterminer à l'avance la probabilité de défaut d'un client. Cette solution s'appuie sur l'emploi de quelques algorithmes utilisés en apprentissage automatique qui sont : la régression logistique, la forêt aléatoire et le Gradient Boosting Machine (GBM).

Contexte

L'évaluation du risque de crédit est une activité centrale pour tout établissement de crédit. Elle est notamment caractérisée par la probabilité de défaillance du client et sa maitrise constitut un atout presque décisif dans la bonne marche du déroulement des activités de l'entreprise. La mesure de ce risque sur les emprunteurs est un enjeu important surtout lorsqu'il s'agit des besoins de financement tel que le petit crédit de la microfinance. Ce risque est en effet lourd de conséquences pour les SFD car toute dette non remboursée est économiquement une perte sèche que supporte le créancier. La nécessité pour les SFD et particulièrement la MDB de disposer d'outils fiables est encore plus importante dans la période actuelle de montée du risque de crédit et de la concurrence. Pouvoir prédire pour un client sur la base de données historiques les probabilités de défaut de remboursement grâce à de l'apprentissage automatique, permettra donc de minimiser le risque de crédit, d'apporter une plus-value au processus d'octroi de crédit et de protéger l'entreprise des problèmes de divers ordres auxquels elle pourrait être exposée suite aux défaillances dans le remboursement des prêts octroyés.

Objectifs

L'objectif principal de ce travail est de proposer un système capable de modéliser le risque de crédit pour une institution de microfinance dénommée MDB et ainsi prédire les probabilités de défaut d'un emprunteur, en utilisant les données historiques sur les crédits individuels. Cet objectif pourra être atteint grâce à la réalisation des objectifs spécifiques qui sont :

- collecter des données historiques de prêt individuel effectué à la MDB ;
- réaliser une analyse exploratoire et un prétraitement des données ;
- construire et évaluer plusieurs modèles d'apprentissage automatique, basé sur différents algorithmes ;
- identifier le modèle le plus performant et le mieux adapté pour la résolution de notre problème ;
- construire une API pour l'utilisation du système final.

Organisation du document

Ce mémoire est subdivisé en trois chapitre :

- le premier chapitre est une revue de littérature où nous exposerons différentes définitions nécessaires pour la compréhension de notre étude et présenterons quelques publications et travaux réalisés qui portent sur la même thématique avant d'y apporter des critiques ;
- le second chapitre présente en détail le flux de travail relatif à notre projet tout en présentant les méthodes et matériels utilisés pour sa réalisation ;
- le troisième chapitre parlera des difficultés rencontrées dans la réalisation de ce travail ainsi que des résultats obtenus et des perspectives qui pourront être explorées pour de futures études.

Présentation de la Mutuelle pour le Développement à la Base(MDB)

La Mutuelle pour le Développement à la Base (MDB) est un Système Financier Décentralisé (SFD) intervenant sur le marché des microfinances, elle a pour principal mission d'offrir des services financiers et non financiers adaptés et fondés sur des principes de solidarité mutuelle en milieu rural et urbain.
En effet, la MDB a le mérite d'être un cas historique qui peut servir de référence au Bénin et en Afrique en ce sens qu'elle est née d'un processus qui a évolué d'un groupe de tontines depuis 1971, passant par des étapes de mutations qui l'ont conduite au statut de Caisse Mutualiste Gibirila Taofic (CMGT) avant de se constituer le 10 mai 1997, en Mutuelle pour le Développement à la Base (MDB) et devenir l'entreprise de référence qu'elle est aujourd'hui (Archive interne de l'entreprise).
La MDB est spécialisée dans l'octroi de crédit et la gestion d'épargne, en ce qui concerne l'octroi de crédit plusieurs produits de crédits sont disponibles, on a le commerce/Artisanat, l'agropastoral, la consommation, le crédit sur tontine et le crédit sur nantissement de DAT, concernant l'épargne on dispose à la MDB des dépôts à vu et des dépôts à terme.

Chapitre 1

Revue de littérature

L'état de l'art constitue une étape utile et importante dans une étude. Elle permet d'avoir une vue complète des solutions et travaux existants, et de porter des jugements objectifs sur l'existant afin de déceler les insuffisances. Nous commençons donc ce chapitre, en définissant quelques thèmes autour desquels tourne notre travail. Nous présentons ensuite, ce qu'est le risque de crédit, les méthodes d'évaluations du risque crédit utilisé à la MDB et finissons par l'étude et la critique de l'existant.

1.1 Définitions

1.1.1 Science des données

En termes généraux, la science des données ou data science en anglais est l'extraction de connaissance d'ensembles de données. La science des donées combine plusieurs domaines, dont les statistiques, les méthodes scientifiques, l'Intelligence Artificielle (IA) et l'analyse des données. Les personnes qui pratiquent la science des données sont appelées « data scientists » et combinent un éventail de compétences pour analyser les données collectées sur le web, les smartphones, les clients, les capteurs et d'autres sources afin d'en tirer des informations exploitables. La science des données englobe la préparation des données (pour l'analyse), le nettoyage, l'agrégation et la manipulation des données pour effectuer une analyse avancée des données.
L'aspect central de la science des données est l'obtention de nouveaux résultats à partir des données : trouver du sens, révéler des problèmes dont vous ne connaissiez pas l'existence et résoudre des problèmes complexes. La science des données repose strictement sur des preuves analytiques, fonctionne avec des données structurées et non structurées et apporte un changement culturel dans les entreprises pour les orienter vers des décisions axées sur les données[11, 7].

1.1.2 Apprentissage automatique

L'apprentissage automatique, également appelé apprentissage machine ou apprentissage artificiel et Machine learning en englais est un champ d'étude de l'Intelligence Artificielle qui se fonde sur des approches mathématiques et statistiques pour donner aux ordinateurs la capacité d'« apprendre » à partir de données.

Elle consiste à laisser des algorithmes découvrir des " patterns ", à savoir des motifs récurrents, dans les ensembles de données. Ces données peuvent être des chiffres, des mots, des images, des statistiques, etc. Tout ce qui peut être stocké numériquement peut servir de données pour le Machine Learning. En décelant les patterns dans ces données, les algorithmes apprennent et améliorent leurs performances dans l'exécution d'une tâche spécifique[9].
L'apprentissage automatique (AA) permet à un système piloté ou assisté par ordinateur comme un programme, une IA ou un robot, d'adapter ses réponses ou comportements aux situations rencontrées, en se fondant sur l'analyse de données empiriques passées issues de bases de données, de capteurs, ou du web.
L'AA (Apprentissage Automatique) permet de surmonter la difficulté qui réside dans le fait que l'ensemble de tous les comportements possibles compte tenu de toutes les entrées possibles devient rapidement trop complexe à décrire et programmer de manière classique (on parle d'explosion combinatoire). On confie donc à des programmes d'AA le soin d'ajuster un modèle pour simplifier cette complexité. Ces programmes, selon leur degré de perfectionnement, intègrent éventuellement des capacités de traitement probabiliste des données, d'analyse de données issues de capteurs, de reconnaissance (reconnaissance vocale, de forme, d'écriture…), de fouille de données, d'informatique théorique, etc[14].
On distingue généralement trois techniques de Machine Learning : l'apprentissage supervisé, l'apprentissage non-supervisé et l'apprentissage par renforcement, mais une approche moins cité existe également, celle de l'apprentissage semi-supervisé.

Apprentissage supervisé Dans le cas de l'apprentissage supervisé, supervised learning en anglais, les données sont étiquetées afin d'indiquer à la machine quelles patterns elle doit rechercher. Le système s'entraîne sur un ensemble de données étiquetées, avec les informations qu'il est censé déterminer. Les données peuvent même être déjà classifiées de la manière dont le système est supposé le faire. Par conséquent, le modèle de Machine Learning sait déjà ce qu'elle doit chercher (patterns) dans ces données. À la fin de l'apprentissage, le modèle ainsi entraîné sera capable de retrouver les mêmes éléments sur des données non étiquetées. Parmi les algorithmes supervisés, on distingue les algorithmes de classification (prédictions non-numériques) et les algorithmes de régression (prédictions numérique). En fonction du problème à résoudre, on utilisera l'un de ces deux types d'algorithme.

- **Les algorithmes de classification** permettent de prévoir des réponses discrètes, par exemple, si un courrier électronique est authentique ou un spam, ou si une tumeur est cancéreuse ou non. Les modèles de classification classent les données d'entrée en catégories. Des applications typiques incluent l'imagerie médicale, la reconnaissance vocale, etc.

- **Les algorithmes de regression** prédisent des réponses continues, par exemple des changements de température ou des fluctuations de la demande de puissance. Des applications typiques incluent la prévision de la charge d'électricité, le trading algorithmique etc.

Apprentissage non supervisé Dans le cas de l'apprentissage non supervisé, unsupervised learning en anglais, les données n'ont pas d'étiquettes. La machine se contente d'explorer les données à la recherche d'éventuelles patterns. Elle ingère de vastes quantités de données, et utilise des algorithmes pour en extraire des caractéristiques pertinentes requises pour étiqueter, trier et classifier les données en temps réel sans intervention humaine. Plutôt que d'automatiser les décisions et les prédictions, cette approche permet d'identifier les patterns et les relations que les humains risquent de ne pas

identifier dans les données. Cette technique n'est pas très populaire, car moins simple à appliquer. Elle est toutefois de plus en plus populaire dans le domaine de la cybersécurité[9].

Apprentissage semi-supervisé L'apprentissage semi-supervisé, semi-supervised learning en anglais, offre un compromis entre apprentissage supervisé et non-supervisé. L'apprentissage semi-supervisé décrit un flux de travail spécifique dans lequel des algorithmes d'apprentissage non supervisé sont utilisés pour générer automatiquement des étiquettes, qui peuvent être introduites dans les algorithmes d'apprentissage supervisé. Pendant l'entraînement, un ensemble de données étiqueté de moindre envergure est utilisé pour guider la classification et l'extraction de caractéristiques à partir d'un ensemble plus large de données non étiquetées[9].
Cette approche s'avère utile dans les situations où le nombre de données étiquetées est insuffisant pour l'entraînement d'un algorithme supervisé, elle permet de contourner le problème.
« Si vous pouvez faire en sorte que les humains étiquettent 0,01 % de vos millions d'échantillons, l'ordinateur peut alors exploiter ces étiquettes pour augmenter de manière significative sa précision prédictive », assure Aaron Kalb, co-fondateur et directeur de la société Alation, une plateforme de catalogue de données d'entreprise.

Apprentissage par renforcement L'apprentissage par renforcement, Reinforcement learning en anglais, consiste à laisser un algorithme apprendre de ses erreurs pour atteindre un objectif. L'algorithme essayera de nombreuses approches différentes pour tenter d'atteindre son but. En fonction de ses performances, il sera récompensé ou pénalisé pour l'inciter à poursuivre dans une voie ou à changer d'approche. Le programmeur fixe les règles pour les récompenses, mais laisse à l'algorithme le soin de décider lui-même des étapes à suivre pour maximiser la récompense, et donc accomplir la tâche qui lui été assigné. Cette technique est notamment utilisée pour permettre à une IA de surpasser les humains dans les jeux.
Par exemple, AlphaGo de Google a battu le champion de Go grâce à l'apprentissage par renforcement. De même, OpenAI a entraîné une IA capable de vaincre les meilleurs joueurs du jeu vidéo Dota 2[9].

1.2 Vue global sur le risque de crédit et son évaluation

1.2.1 Définition

D'une manière générale le risque de crédit aussi appelé le risque de contrepartie est le risque qu'un emprunteur ne rembourse pas tout ou une partie de son crédit aux échéances prévues par le contrat signé entre lui et l'organisme préteur[15].
Dès qu'un agent économique consent un crédit à une contrepartie, une relation risquée s'instaure entre le créancier et son débiteur. Ce dernier peut en effet, de bonne ou de mauvaise foi, ne pas payer sa dette à l'échéance convenue. L'aléa qui pèse sur le respect d'un engagement de régler une dette constitue le risque de Crédit. Il est caractérisé par la probabilité de défaillance du client et est inhérent à l'activité d'intermédiation que les organismes préteurs jouent dans le financement de l'économie. D'après Henri calvet (1997) « Le risque de contrepartie peut être défini comme étant « Un risque de perte lié à la défaillance d'un débiteur sur lequel l'établissement de crédit détient un crédit ». Ceci correspond à une défaillance possible des agents avec lesquels les organismes preteurs

se sont engagées et qui constituent les contreparties. Une telle défaillance peut se traduire par le non remboursement de crédits par des emprunteurs privés nationaux en difficulté ou par le non transfert du remboursement des crédits accordés en devises à des non-résidents. Il s'assimile au degré d'incertitude qui pèse sur l'aptitude d'un emprunteur à effectuer le service prévu de la dette, c'est-à-dire à l'incertitude des pertes pouvant être générées par un crédit à un créancier financier.
Au vue de l'importance que ce risque revêt les prêteurs doivent mesurer avec précision le risque de crédit des emprunteurs avant de leur accorder un crédit et de fixer les conditions de son octroi (montant, maturité, taux et covenants), plusieurs techniques sont alors utilisées pour l'évaluation du risque de crédit[5] :

- **Les méthodes postives** : Le principe fondateur de ces méthodes est de traiter et observer un ensemble de données pour en déduire une appréciation du risque d'une entreprise, issue d'un constat subjectif, plus ou moins justifié. Cette approche est largement descriptive et n'aboutissent pas à un indicateur de synthèse pouvant s'interpréter en termes de risque de défaut ou de faillite. On distingue parmi les méthodes positives, l'analyse financière, la méthode des 5C, la méthode LAPP.

- **Le rating** : La notation « Rating » c'est un mot d'origine américain qui veut dire évaluation. El karyotis, 1995 définit la notation comme : « un processus d'évaluation de risque attaché à un titre de créance, synthétisé à une note, permettant un classement en fonction des caractéristiques particulières du titre proposé et des garanties offertes par l'émetteur. ». La notation financière est l'expression de la solvabilité d'un emprunteur, elle mesure la capacité de ce dernier à rembourser toutes les sommes dues à court ou à long terme. La notation financière se concrétise par différents types de notation, elle est soit attribuée par des sociétés spécialisées de notation, on parle donc de notation externe, soit établie par l'institution financière elles-mêmes, la notation alors est dites internes.

- **Le Scoring** : Cette technique est définit par (Mester, 1997) comme « une méthode statistique pour prédire la probabilité qu'un demandeur de prêt (débiteur) fasse défaut ». Le credit scoring permet de mesurer la probabilité de défaut sur les crédits proposés aux particuliers et aux professionnels. Cette technique peut se baser soit sur des données historiques ou sur des variables statistiques. Les informations de l'emprunteur constituent une base pour connaitre ses caractéristiques et prévoir si celui-ci aura une solvabilité future. Les établissements de crédit peuvent ainsi classer les débiteurs en fonction de la proportion du risque. Cette analyse est plus réservée à une clientèle de particuliers et de petites entreprises. La relation de ces emprunteurs est moins complexe que les grandes entreprises.

1.2.2 Méthode d'évaluation du risque de crédit utilisée à la MDB

La solvabilité du client s'évalue à la MDB à travers l'étude des dossiers de demande de crédit introduit par les clients auprès de l'institution. Cette étude est conduite respectivement par les agents de crédits, le comité d'agence, le comité technique et le comité de crédit.
Les agents de crédit débutent l'analyse de tout dossier de crédit introduite auprès de la MDB. Mais, parallèlement à cette analyse des dossiers de demande de crédit, les agents de crédit assurent le conseil et l'orientation du client afin de lui permettre de bien circonscrire l'objet de sa demande. Cette analyse consiste notamment à réaliser une analyse financière à travers le calcule et l'interprétation de plusieurs ratio, le ratio de rentabilité, le ratio d'autonomie financière et le ratio de remboursement qui

constitut le principal ratio utilisé par les analystes de la MDB. En complément à l'analyse financière réaliser plusieurs autres actions sont également entrepris il s'agit de :

- faire une description des caractéristiques du client et de son besoin exprimé ;
- faire l'état de la situation des engagements en cours du client vis-à-vis de la MDB et d'autres structures ;
- recenser les types de garanties que le client propose ;
- résumer les points forts et les points faibles susceptibles d'orienter une appréciation du dossier en traitement ;
- faire une proposition de décision vis-à-vis du financement sollicité par le client ;
- effectuer une descente sur le terrain afin de mesurer la véracité des informations transmise par le client lors de l'entretient.

Le dossier ainsi étudié par les analystes est ensuite envoyé aux diverses instances compétentes pour une validation successive.

1.3 Etat de l'art

1.3.1 Publications scientifiques

Plusieurs articles scientifiques traitent du sujet de l'évaluation du risque de crédit avec de l'apprentissage automatique, quelques uns d'entre eux ont été abordé dans notre travail :

1. **Asad Mumtaz** dans son article[10] présente une approche basée sur l'apprentissage automatique pour la modélisation du risque de crédit. Le modèle conçu a obtenu un score de 0,866 pour le AUROC et de 0,732 pour le Coefficient de Gini. L'ensemble de données utilisés est disponible sur la plateforme kaggle[8] et concerne les prêts à la consommation émis par le Lending Club, un prêteur P2P américain, Les données brutes comprennent des informations sur plus de 450 000 prêts à la consommation émis entre 2007 et 2014 avec près de 75 caractéristiques, y compris le statut actuel du prêt.
   Après les étapes d'exploration et fractionnement des données, de nettoyage des données, de selection et d'ingénieurie de fonctionnalité, un modèle de régression logistique à été ajuster sur l'ensemble de données d'apprentissage, l'un des objectifs du travail présenté dans cette article étant le développement d'une carte de pointage à la suite de la formation du modèle du risque crédit, tout les attributs (variables prédictives) de l'ensemble de donnée ont dû adopter une nature catégorielle, de nouveaux attributs catégorielles pour toutes les variables numériques et catégorielles ont donc été créer en s'appuyant sur le concept du poids de la preuve (WoE) ce qui a impliqué une discrétisation des caractéristiques numériques, la valeur de l'information (IV) à été ensuite utilisé pour aider à classer les fonctionnalités en fonction de leurs importance relative.

2. **Raphaël Bastos** dans son article[6], construit un modèle de machine learning pour prédire le risque de défaillance des clients pour une fintech brésilienne en utilisant une approche basée sur l'apprentissage automatique. Le jeu de donnée utilisé provient de Nubank, une banque numérique brésilienne et l'une des plus grandes Fintech d'Amérique latine. Le travail est effectué sur la base d'un ensemble de données contenant 43 fonctionnalités pour 45 000 clients. Une analyse exploratoire un prétraitement des données on été au préalable éffectuer pour une meilleur compréhension des données et leur transformation dans un format propice au fonctionnement d'un modèle de machine learning. Pour la construction du modèle de machine learning trois algorithmes d'apprentissage automatiques, ont été utilisés pour ce travail, il s'agit d'Algorithme d'amplification de gradient tel que : XGBoost, LightGBM et CatBoost.
La métrique utilisée pour l'évaluation des modèles est le "recall". Pour de meilleurs résultats une validation croisée 5 fois a été également effectué.

	Recall
XGBClassifier	0.662726
LGBMClassifier	0.647915
CatBoostClassifier	0.645911

FIGURE 1.1 : Valeur du recall pour chaque modèle construit [6]

.

Après le réglage des hyperparamètres à l'aide de GridSearchCV (de la bibliothèque python "sklearn"), pour la recherche de valeurs des paramètres, et évaluations réaliser sur l'ensemble de test , il ressort que le modèle avec l'algorithme XGBoost a donné les meilleurs résultats, avec un taux de rappel de 81%, bien qu'il ait généré 56% de faux positifs indésirables.

1.3.2 Présentation de quelques logiciels intélligents d'évaluation du risque de crédit

- **Présentation de Zest**

  Zest est un logiciel de la société Zest AI qui utilise l'apprentissage automatique et des milliers de points de données pour évaluer le risque d'un emprunteur. Zest permet aux utilisateurs de déployer et de gérer des modèles de crédit, il est utilisé pour le traitement de l'information, la modélisation, la documentation et le déploiement, il offre également d'autres services tel que l'aide à la souscription de crédit, l'identification de nouveaux emprunteurs etc. Zest utilise une technique appelée "débiaisage contradictoire" où une IA crée un modèle analytique, et une autre critique ce modèle pour ses biais. La première IA ajuste alors ses modèles en fonction des biais identifiés par l'IA critique. De manière plus concrète deux modèles d'apprentissage automatique sont opposés, l'un tente de prédire la solvabilité tandis que l'autre devine la race, le sexe et d'autres attributs du candidat notés par le premier modèle. La concurrence pousse les deux à améliorer leurs méthodes jusqu'à ce que le prédicteur ne puisse plus distinguer les résultats de race ou de sexe du premier modèle, ce qui donne un modèle apparemment plus précis et équitable[13].

- **Présentation du logiciel de Underwrite.ai**

  Le système de Underwrite.ai applique les avancées de l'intelligence artificielle dérivées de la génomique et de la physique des particules pour fournir aux prêteurs des modèles dynamiques et non linéaire de risque de crédit, en utilisant les techniques de l'apprentissage automatique. Underwrite.ai utilise la modélisation algorithmique non linéaire pour déterminer efficacement le risque de crédit dans les pays où l'utilisation des bureaux de crédit est très limitée ou inexistante. Underwrite.ai utilise une multitude d'algorithmes génétiques, de réseaux de neurones, de forêts aléatoires, de machines d'amplification de gradient. La technique de Underwrite.ai consiste à isoler tous les attributs avec une valeur prédictive dans l'ensemble d'apprentissage, Il s'agit généralement d'un sous-ensemble raisonnablement important d'attributs, généralement plus de 1 000, ensuite commence le processus de sélection des caractéristiques, qui consiste à combiner aléatoirement ces attributs en éléments combinatoires et à tester chaque combinaison pour la précision prédictive[12].

1.3.3 Critique de l'existant

La critique de l'existant est une étape importante après l'étude de l'existant. Elle a pour but de porter un jugement objectif sur l'existant. Face aux différentes solutions existantes, nous avons relevé quelques insuffisances à savoir :

- le traitement des données est liée aux politiques de gestion de chaque société ;
- aucune de ses solutions ne prend en charge le nettoyage personnalisé (propre à chaque société) préalable des données avant utilisation ;
- certaines techniques utilisées dans la mise en place de certaine solution, sont assez laborieuses ;
- les logiciels présentés sont des outils payants ;
- chaque chercheur propose sa méthodologie.

Conclusion

Tout au long de ce chapitre nous avons abordé des notions liées à l'apprentissage automatique et au risque de crédit. Nous avons exposé les différentes méthodes utilisées généralement pour l'évaluation du risque de crédit dans les institutions financières. Pour finir nous avons présenté plusieurs travaux de recherche et outils mis en place pour la modélisation du risque de crédit grâces à l'apprentissage automatique. Le chapitre suivant fera l'objet de la description du matériel et des méthodes utilisés tout au long de notre travail.

Chapitre 2

Matériel et méthodes

Dans ce chapitre, nous faisons la description du matériel utilisé tout au long de nos travaux ainsi que les outils de travail. Nous détaillons ensuite la manière dont nous avons préparé notre environnement de travail, de même que les étapes de l'implémentation.

2.1 Matériel

2.1.1 Kit système

Pour la réalisation de notre travail nous avons utilisé un ordinateur portatif dont les caractéristiques sont les suivantes :

- **Modèle** : Inspiron 15 3000 - Dell
- **Système d'exploitation** : Windows 10
- **Architecture** : 64bits
- **Processeur** : Intel(R) Core(TM) i3-8145U CPU @ 2.10GHz 2.30 GHz
- **Mémoire RAM** : 8 Go

2.1.2 Kit pour la conception du modèle de machine learning et de l'API

Langage de programmation Python : est un langage de programmation interprété, multi-paradigme et multiplateformes. Python est très sollicité par une large communauté de développeurs et de programmeurs, cest un langage simple, facile à apprendre et qui permet une bonne réduction du cout de la maintenance des codes. Les bibliothèques python encouragent la modularité et la réutilisabilité des codes. En ce qui concerne le domaine de l'apprentissage automatique, Python se démarque notamment en proposant une pléthore de bibliothèques de haute qualité, couvrant tous les types d'apprentissage disponibles sur le marché, le tout accompagné d'une large communauté dynamique.

Anaconda : Anaconda est une distribution libre et open source des langages de programmation Python et R appliqué au développement d'applications dédiées à la science des données. Elle comprend

une série d'applications, de bibliothèques et de concepts conçus pour les travaux de science des données. Anaconda fonctionne comme un gestionnaire d'environnements, un gestionnaire de packages et a une collection de plus de 720 packages open source.

Jupyter notbook : est une application client-serveur qui permet de modifier et d'exécuter des cahiers électroniques (notebook documents) via un navigateur Web sans nécessairement un accès à internet. Les cahiers électroniques produit par l'application jupyter notbook peuvent contenir à la fois du texte, des images, des formules mathématiques et du code informatique exécutable. Les notebook documents sont à la fois des documents lisibles par l'homme contenant la description de l'analyse et les résultats (figures, tableaux,texte etc.) ainsi que des documents exécutables qui peuvent être exécutés pour effectuer l'analyse des données.

Scikit-learn : est une bibliothèque libre Python dédiée à l'apprentissage automatique. Elle comporte divers algorithmes de classification , de régression et de clustering. Elle est conçu pour interagir avec les bibliothèques numériques et scientifiques Python NumPy et Scipy.

Flask : Flask est un micro-framework web écrit en Python. Il vous permet de concevoir une application web solide rapidement.

Postman : est une application permettant de tester des API

2.1.3 Messure de performance

Afin de pouvoir évaluer nos performances tout au long de l'étude on décide de considérer les métriques que sont la matrice de confusion, la justesse, la précision, le rappel, le score F-1 et le score AUC.

1. **Matrice de confusion** : La matrice de confusion nous aide à visualiser les erreurs commises par le modèle sur chacune des classes, qu'elles soient positives ou négatives. Par conséquent, cela nous renseigne sur les erreurs de classification pour les deux classes.

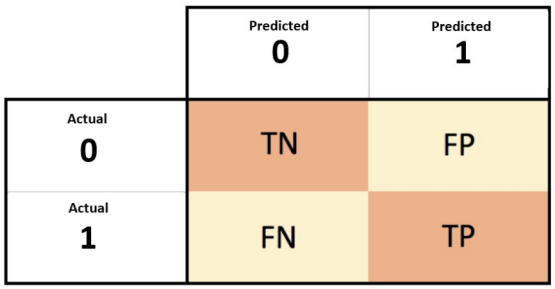

FIGURE 2.1 : Matrice de confusion

- **TN (True Negative)** : il s'agit du nombre de résultats initialement négatifs (0) et prédits négatifs (0).
- **FP (False Positive)** : Il s'agit du nombre de résultats initialement négatifs (0) mais dont la prédiction est positive (1).
- **FN (False Negative)** : Il s'agit du nombre de résultats initialement positif (1) mais dont la prédiction est negative (0).

- **TP (True Positive)** : il s'agit du nombre de résultats initialement positif (1) et prédits positif (1).

2. **Justesse (Accuracy)** : c'est la proportion du nombre total de prédictions qui étaient correctes.

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN}$$

3. **Précision (Precision)** : Elle indique la proportion de cas positifs prédit correctement, dans l'ensemble des prédictions positives.

$$Precision = \frac{TP}{TP + FP}$$

4. **Rappel (Recall)** : Elle vise à mesurer la proportion de cas positifs réels qui a été correctement identifiée dans l'ensemble des cas positifs réels.

$$Recall = \frac{TP}{TP + FN}$$

5. **Score F1 (F1 Score)** : Il indique la moyenne harmonique de Précision et Rappel.

$$F1Score = \frac{2 \times (Precision \times Recall)}{Precision + Recall}$$

6. **L'AUC** : L'AUC en français zone sous la courbe ROC (Area under the ROC Curve) est une mesure globale des performances pour tous les seuils de classification possibles du modèle, Pour un modèle idéal, on a AUC=1, pour un modèle aléatoire (Aucun pouvoir discriminant), on a AUC=0.5, On considère habituellement que le modèle est bon dès lors que la valeur de l'AUC est supérieure à 0.7.

2.2 Méthodes

Ce projet a été réalisé en suivant la méthodologie Scrum-Agile. C'est un ensemble de principes de développement logiciel ayant pour but de bien gérer la conception, l'implémentation et la mise en place d'un produit ou d'une solution informatique. Cette méthodologie décrit une strategie flexible de développement où les developpeurs travaillent sur différents modules permettant d'obtenir le produit final. L'avantage de la méthode Scrum est qu'elle met au coeur de tout le processus de réalisation l'utilisateur final de la solution en cours de développement et permet de s'adapter plus facilement aux modifications ou changements pouvant survenir pendant l'éxecution du projet.

Tout au long de l'étude, les différentes phases pour l'apprentissage, l'évaluation[1] et le déploiement du modèle, de manière générale sont résumées suivant les étapes de la figure 2.2 suivante :

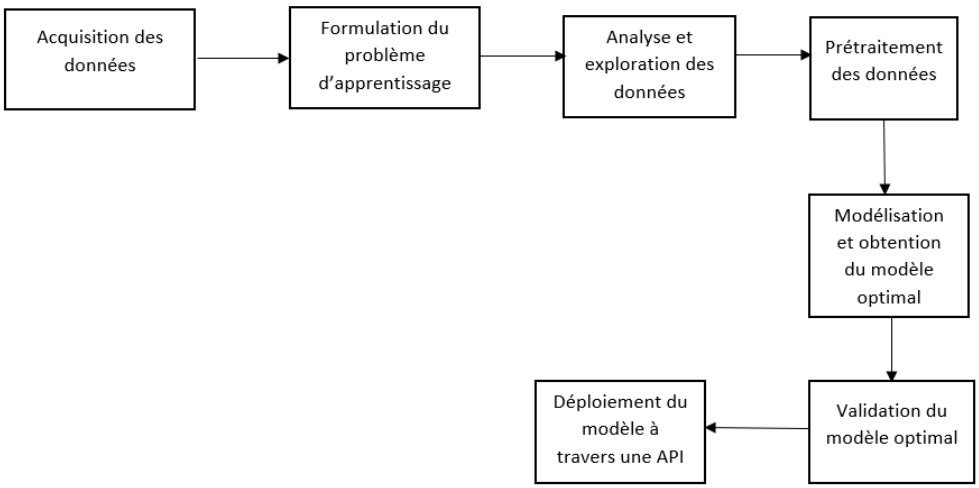

FIGURE 2.2 : Phases d'apprentissage, d'évaluation et de déploiement du modèle

2.2.1 Acquisition des données

L'ensemble de données utilisé pour ce travail est issu de la fusion de trois tables relationnelles, importés depuis la base de données rattaché à l'application métier de la MDB. Ces tables contiennent des données sur les demandes de prêt effectués à au niveau de l'agence de Cotonou de la MDB sur une période allant de 1995 à 2021.

- **La table "ADHERENTS"** : Cette table est dotée de 8598 lignes et de 48 colonnes, elle contient des informations personnelles (Nom, mail, ville, date de naissance, etc.) du client et d'autres informations recueillies lors de l'inscription de ce dernier. Chaque client est identifié par son 'COD_ADH' (Code adhérent).

- **La table "DEMPRET"** : Cette table est dotée de 7476 lignes et de 44 colonnes, elle contient des informations relatives à une demande de prêt (produit de crédit, montant de prêt sollicité, période de remboursement, etc.), chaque demande de prêt est identifiée par 'REF_DEMANDE' (référence de la demande), la table "ADHERENTS" est liée à la table "DEMPRET" par la clé étrangère 'COD_ADH'. Dans cette table un client peut avoir plusieurs demandes de prêt .

- **La table "PRETS"** : Cette table est dotée de 7128 lignes et de 32 colonnes, elle contient des informations relatives à chaque prêt notamment le statut du prêt (soldé ou perte), le montant du prêt, le nombre d'échéances, etc. Chaque prêt est identifié par ''NUM_DOSSIER" (Numéro de dossier), dans cette table pour une demande de prêt on a un seul prêt pris en charge. La table "PRET" est liée à la table "DEMPRET" par la clé étrangère 'REF_DEMANDE'

La fusion des trois tables ADHERENTS, DEMPRET, PRET, pour l'obtention de la table finale s'est faite grâce à l'opérateur de jointure de l'algèbre relationnelle. Les opérations de jointure effectuées avec les trois tables ce sont déroulés en deux étapes :

- La première étape a consister à combiner la table ADHERENT et la table DEMPRET en utilisant une jointure externe complète (FULL OUTER JOIN), le choix d'une telle jointure à été fait pour conserver le maximum de données, la combinaison des tables a été fait suivant la colonne 'COD_ADH' (se trouvant dans les deux tables).

- La table issue de la première étape a été combiné à la table "PRETS" grâce à une opération de jointure interne (INNER JOIN), qui s'est faite suivant la colonne REF_DEMANDE. Ici le choix

d'une jointure interne a été fait pour que uniquement les individus ayant fait un prêt puissent être conservés dans la table finale puisque seuls les clients ayant fait un prêt se retrouvent dans la table "PRETS".

Avant la combinaison des tables ces derniers ont été convertis en fichier csv puis enregistré dans des objets python de type DataFrame (Structure de type table SQL), les opérations de jointure ont également été effectué en python grâce à la méthode "merge ()" qui permet de fusionner des objets de type DataFrame ou Series avec des jointures de type base de données. L'ensemble de données final a été enregistré dans un objet python de type DataFrame.
Notre ensemble de données final est un ensemble de données déséquilibré, où la classe négative(Crédit soldé) domine la classe positive(Crédit en perte), car il n'y a qu'un petit nombre de défaillants parmi tous les candidats, il faut également retenir que les données obtenues ont été anonymisé avant toute manipulation.

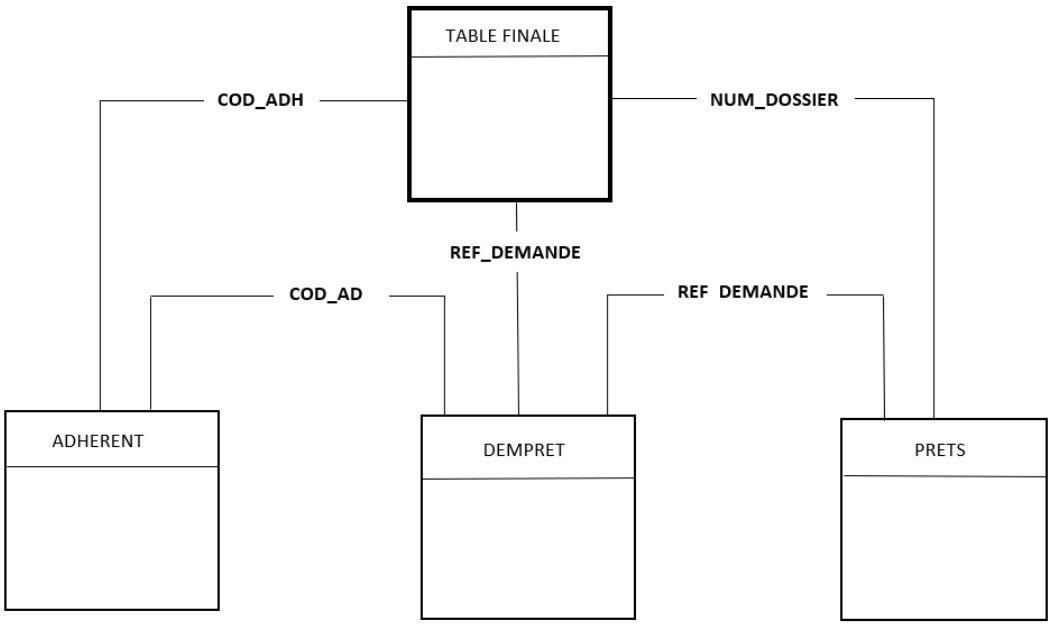

FIGURE 2.3 : Structure des tables relationnelles

2.2.2 Formulation du problème d'apprentissage

Avant de formuler le problème d'apprentissage, il est important d'identifier les objectifs et les contraintes de l'entreprise :

Objectifs :

- L'objectif principal est d'identifier les défaillants potentiels sur la base des données fournies sur les candidats.
- La probabilité de classification est essentielle car nous voulons être très sûr lorsque nous classons quelqu'un comme non défaillant, car le coût d'une erreur peut être très élevé pour l'entreprise.

Contraintes :

- L'interprétabilité est partiellement importante pour classer quelqu'un comme défaillant ou non.
- Il n'y a pas d'exigence stricte de latence, car l'objectif est plus de prendre la bonne décision que de prendre une décision rapide. Ce serait bien et acceptable si le modèle prenait quelques secondes pour faire une prédiction.

Problème d'apprentissage :

Après avoir identifié les objectifs et les contraintes de l'entreprise, nous pouvons maintenant formuler l'énoncé du problème d'apprentissage automatique, qui respecterait ces objectifs et contraintes :

- nous avons identifier qu'il s'agit d'un problème d'apprentissage supervisé, ici, les étiquettes de classe indiquent si un candidat donné est un défaillant ou non. Ainsi, pour un client emprunteur donné, en utilisant les caractéristiques données, nous devons prédire l'étiquette de classe ;
- nous réalisons également qu'il s'agit d'un problème de classification binaire, c'est-à-dire qu'il ne contient que 2 classes, à savoir Positif (1) et Négatif (0) ;
- l'ensemble de données est très déséquilibré.

Une fois que nous avons construit le modèle d'apprentissage automatique final, nous pouvons ensuite le déployer pour vérifier instantanément les risques potentiels associés au prêts d'un client , qui peut être sois un nouveau ou un ancien client de la MDB.

2.2.3 Analyse et exploration des données

L'analyse exploratoire des données est une étape importante dans la réalisation d'un projet d'apprentissage automatique, elle fait référence au processus consistant à effectuer des enquêtes initiales sur les données afin de découvrir des modèles, de repérer des anomalies, de tester des hypothèses et de vérifier des hypothèses à l'aide de statistiques récapitulatives et de représentations graphiques.

a) **Statistique de base**

Dans notre ensemble de donnée nous avons 7240 observations (lignes de donnée) et 122 attributs (colonnes). Nous avons 63 variables de type float, 41 variables de type object et 18 variables de type int. Dans le contexte de l'apprentissage automatique dans lequel nous somme nos variables prédictives sont aussi appelé fonctionnalités ou features en anglais, notre variable cible peut être aussi appelé target en anglais.

b) **Nettoyage des données (Data Cleaning)**

- **Valeurs manquantes et doublons**

  Nous avons beaucoup de valeurs manquantes, mais on remarque que les caractérisques les plus importantes (ayant une forte probabilité de révélé des informations utiles) ne contiennent pratiquement pas de données manquantes, ce qui est une bonne chose. Un nouveau dataframe a donc été mis en place sans les colonnes avec plus 70% de valeurs

manquantes, les colonnes les plus prometteuses faisant partie de cette catégorie sont également supprimées car essayer de les imputer pourrait conduire à des inexactitudes, le nouvel ensemble de données comporte donc 77 colonnes. En ce qui concerne les doublons on a retrouvé dans l'ensemble de données deux observations avec la même référence de demande de prêt (REF_DEMANDE), ce qui n'est pas normal car chaque prêt a une référence de demande unique, nous allons donc les supprimer.

- **Autres anomalies**

  Afin d'éviter que les resultats de l'analyse de données ne soit biaisés des corrections on été apportées à l'ensemble de données pour respecter la politique de l'entreprise et pour corriger quelques anomalies : ces modifications étaient éssentiellement : l'anonymisation des données, la convertion de certaines variables en variable avec des types adaptés ; la correction de certaines erreurs humaines (Données mal renseignés) et l'élimination des observations non étiquetées avec "SD"(soldé) et "PE"(perte) et des observations concernant des credits demandés par un groupe de personne ou une société ; la suppression du produit de crédit ayant code de produit de crédit '3.0'(Prêts aux plus démunies), car ce produit de crédit n'a jamais été accordé et enfin, le remplacement des taux d'intérêt à 0% sur les crédits par les taux d'intérêt appropriés.

c) **Répartition des variables**

(a) **La variable cible (ETAT_PRET)**

Après avoir examiné la distribution de la variable cible dans notre ensemble de données. Nous avons constaté que uniquement 0.09 % des clients était des défaillants et que les 0.91 % restant constuaient la proportion des clients non-défaillants dans l'ensemble de données (Voir figure 2.4). Cela montre que la classe Positive (perte) est une classe minoritaire dans notre jeu de données. Cela implique également qu'il s'agit d'un ensemble de données déséquilibré et que nous devons trouver des moyens adéquats pour le gérer.

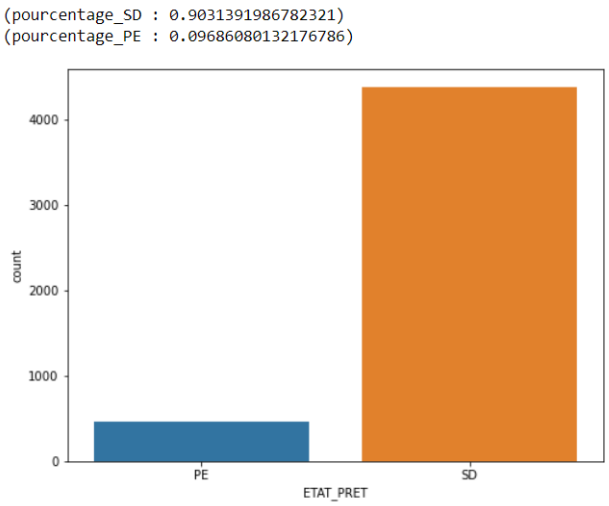

FIGURE 2.4 : Distribution de la variable cible (ETAT_PRET)

(b) **Les variables continues et catégorielles**

Pour les variables continues, nous avons largement utilisé les Box-Plots, et les histogrammes,

pour les variables catégorielles ce sont les diagrammes à bart qui ont été utilisé. Nous avons tracé la distribution des points de données pour les défaillants et les non-défaillants ensemble, et avons essayé de voir si ces distributions sont similaires ou se distinguent afin de tirer des conclusions ; à ces graphiques, a eté ajouté un résumé statistique de chaque variable.

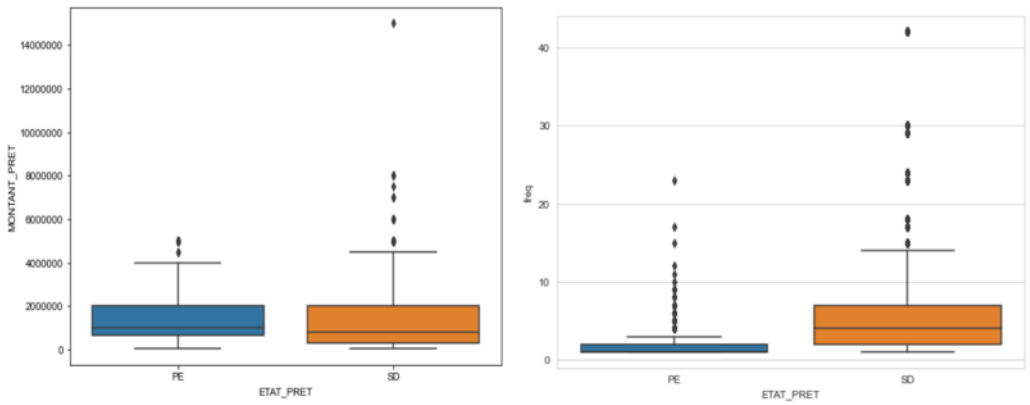

FIGURE 2.5 : Distribution des variables MONTANT_PRET (montant du Prêt), et freq (fréquence d'emprunt) en fonction des statut ou en perte du Prêt

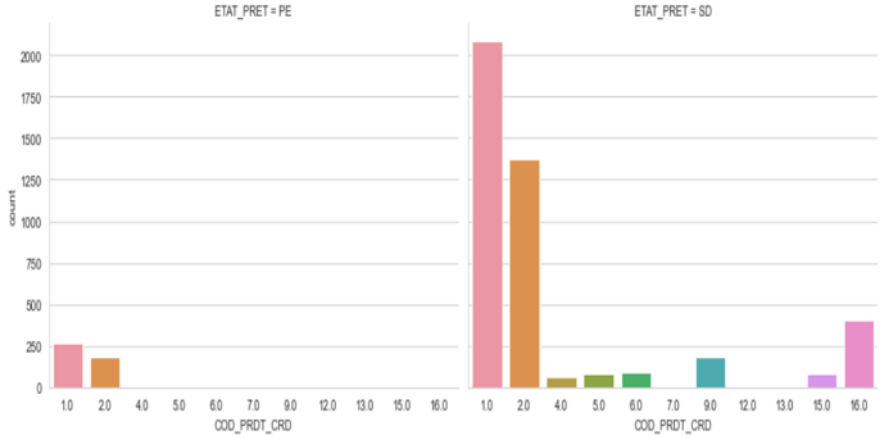

FIGURE 2.6 : Distribution de la variables COD_PRDT_CRD (code du produit de crédit) en fonction des statut ou en perte du Prêt

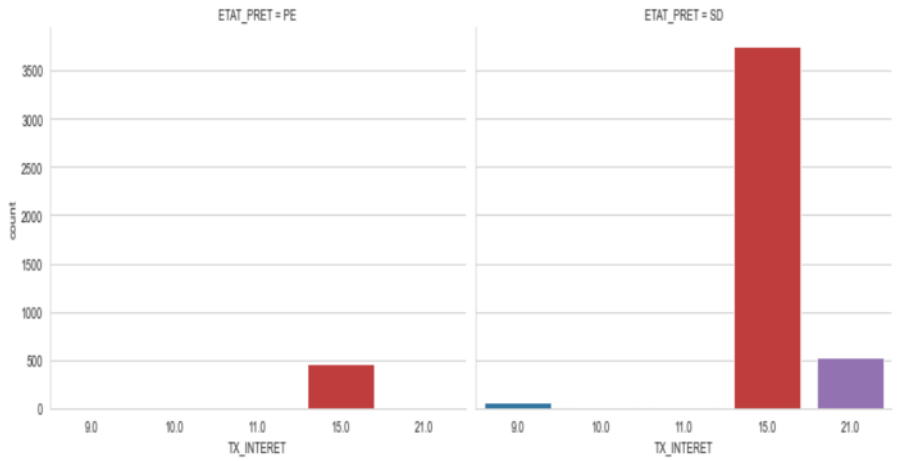

FIGURE 2.7 : Distribution de la variables TX_INTERET (taux d'intérêt sur un Prêt) en fonction des statut soldé ou en perte du Prêt

	MONTANT_PRET	MONTANT_SOLLICITE	DUREE_GRACE	DUREE_GRACE_SOLLICITE	AGE	freq
count	4842.0	4842.0	4842.0	4842.0	4842.0	4842.0
mean	1 279 990.0	1 576 759.0	0.0	0.0	45.0	5.52
std	1 303 475.0	1 600 033.0	0.0	0.0	11.0	6.06
min	45 000.0	45 000.0	0.0	0.0	0.0	1.0
25%	300 000.0	300 000.0	0.0	0.0	37.0	2.0
50%	900 000.0	1000 000.0	0.0	0.0	45.0	4.0
75%	2 000 000.0	2 000 000.0	0.0	0.0	53.0	7.0
max	15 000 000.0	15 000 000.0	0.0	0.0	79.0	42.0

TABLE 2.1 : Statistiques descriptives de quelques variables continues

	COD_TYPEADH	VILLE
count	4842	4209
unique	1	12
top	PP	COTONOU
freq	4842	3604

TABLE 2.2 : Statistiques descriptives de quelques variables catégorielles

Grâce à l'étude des graphiques et du résumé statistique de chaque variable nous avons pu observer de nombreuse remarques pertinentes :

- une grande différence entre les moyennes et les écart-type d'une variable à une autre dans notre ensemble de données ce qui indique que les données ne sont pas à la même échelle et qu'il faudra probablement normaliser les données pour un meilleur résultat ;
- la présence de valeurs manquantes au niveau de la fonctionnalité 'VILLE' et d'éventuelles valeurs aberrantes au niveau des fonctionnalités intitulé 'MONTANT_PRET' et 'MONTANT_SOLLICITE' du fait de la différence importante entre leur 3ème quantile et leur maximum, les valeurs aberrantes sont très distantes des autres valeurs et ne sont donc pas représentatives de l'ensemble de données, elles peuvent causer d'importantes erreurs de modélisation (Voir tableaux 3.1 et 2.2) ;
- la présence de variables dont les valeurs ne varient pas, donc des fonctionnalités sans intérêt, il s'agit de certaines fonctionnalités intitulés 'DECISION', 'NBRE_BENEF', 'DUREE_GRACE', 'DUREE_GRACE_SOLLICITE', 'COD_TYPEADH' (Voir tableau 3.1) ;
- on remarque également de grande tendance dans les données, notamment un montant d'emprunt et une fréquence d'emprunt plus élevée chez les clients en défaut (Voir figure 2.5) mais aussi une quasi complète concentration des prêts de personnes en défaut autour des produits de credit 1 et 2 (prêts aux salariés/Fonctionnaire et prêts aux commerçants et artisans) qui sont les produits ayant le plus fort taux d'intérêt, 15% (Voir figure 2.6 et 2.7) ;

Cet étude des graphiques et du résumé statistique de chaque variable nous a permis de retrouver les grandes tendances dans les données et de visualiser les variations des valeurs de nos variables en fonction du statut soldé ou en perte du prêt pour un client. Nous avons également procédé à une ingénierie de fonctionnalités qui a consisté à créer de nouvelles variables (age, fréquence d'emprunt et encienneté) à partir des anciennes variables, pour une analyse plus poussée.

2.2.4 Prétraitement des données (Data Preprocessing)

L'objectif de la phase de prétraitement des données (preprocessing) est de Transformer l'ensemble de données pour le mettre dans un format propice au fonctionnement d'un modèle de machine learning. La stratégie utilisée pour le prétraitement des données est basée sur les conclusions de la phase d'analyse et d'exploration des données. Les étapes de cette partie de notre travail ont consisté essentiellement en :

- **La suppression de variable** : les variables déclarées comme inutiles ou non informatives et ceux fortement corrélées à d'autres variables ont été supprimé.
- **La gestion des valeurs aberrantes (outliers)** : la méthode quantile() de la bibliothèque pandas a été utilisé pour découvrir quelle est la plage de la quantité majoritaire des données (entre 0,05 centile et 0,95 centile dans notre cas), toutes les valeurs inférieures à la limite inférieure de cette plage ont été arrondi à la limite inférieure, de même les valeurs au-dessus de la limite supérieure de cette plage ont été arrondi à cette limite supérieure.
- **Gestion des valeurs manquantes** : ici une seule variable était concernée (la variable ville), et au vu de la nature catégorielle de cette dernière toutes les valeurs manquantes ont été imputé avec la catégorie "Cotonou" qui est la catégorie ayant la plus forte occurrence au sein de la colonne de cette variable.
- **Encodage des variables catégorielles** : Les catégories de nos variables catégorielles au format non numérique n'étant pas nombreuses, un dictionnaire python a été créé pour faire correspondre les valeurs de nos variables à des entiers, la logique au sein de ce dictionnaire a été ensuite appliqué à nos variables grâce à une fonction de mapping. L'encodage des données est nécessaire pour mettre les données dans un format propice au fonctionnement d'un modèle d'apprentissage automatique.
- **Fractionnement des données** : l'ensemble de données obtenu a été ensuite divisé en ensemble d'entrainement (données avec lesquelles le modèle sera entrainé) et en ensemble de test (données sur lesquelles le modèle sera testé) de manière à reproduire la répartion des classes(0,1) observé dans l'ensemble de données d'origine, chacun de ces ensembles représente respectivement 80% et 20% de l'ensemble de données d'origine. Nos ensembles de données d'entrainement et de test comprennent respectivement 3873 lignes de données avec 14 colonnes et 969 avec 14 colonnes.
- **Pipeline de prétraitement des données** : Au cour de cette phase beaucoup de test avec des combinaisons de paramètre ont été réalisés pour la sélection des meilleurs éléments pour nos piplines de prétraitement de données. Les éléments finalement sélectionnés sont les méthodes MinMaxScaler() pour la mise à l'échelle des données, SelectKbest(f_classif, k = 10) pour la sélection des dix meilleures variables prédictives avec la mesure statistique ANOVA F et la technique de suréchantillonnage synthétique des minorités ou SMOTE() qui a servi pour régler le problème de déséquilibre de classe dans notre ensemble de données.

2.2.5 Modélisation et obtention du modèle finale

Pour notre travail nous avons utilisé trois algorithmes d'apprentissage automatique, les trois algorithmes ont ensuite servi à construire trois modèles de machine learning dont les performaces on été

ensuite comparé afin d'otenir le modèle le plus optimal, les trois algrithmes utilisés sont : logistic regression (regression logistique), Random Forest classifier (forêts d'arbres décisionnels) et Gradient Boosting Machine (amplification de gradient). Notre choix s'est porté sur ces algorithmes car nous cherchions pour notre problème à comparer les perfomances d'algorithmes fortement basés sur le concept de probabilité et d'algorithme utilisant les méthodes de bagging et de boosting. La regression logistique a été particulièrement choisi car c'est un algorithme couramment utilisé dans l'évaluation du risque de crédit, l'algorithme de random forest a été choisi car il utilise la méthode de bagging et est très résistant au problème de sur-ajustement, quant à l'algorithme de gradient boosting machine il a été choisi car c'est un algorithme d'amplification de gratient très utilisé dans les problèmes de classification et également résistant au sur-ajustement.

a) **Logistic regression (regression logistique)**
La régression logistique est un algorithme d'apprentissage automatique qui est utilisé pour les problèmes de classification, c'est un algorithme d'analyse prédictive et basé sur le concept de probabilité, elle utilise une fonction de coût définit comme la fonction logistique (aussi appelé fonction sigmoïde ou tout simplement sigma σ). Cette fonction a la particularité d'être toujours comprise entre 0 et 1 [4].

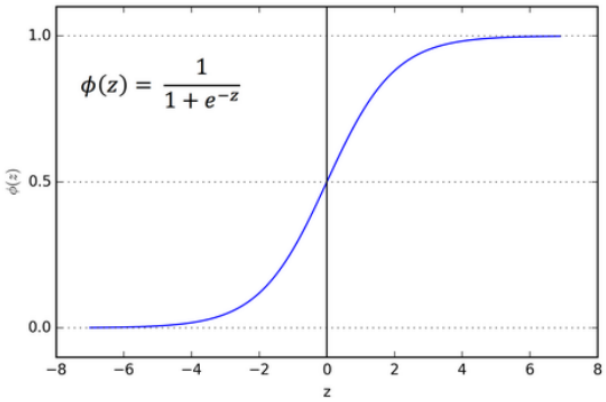

FIGURE 2.8 : Fonction logistique

Pour ajuster la fonction logistique sur un Dataset (X,y) on y fait passer le produit matriciel $X.\theta$ (avec X la matrice contenant nos variables prédictives, θ la matrice contenant les paramètres de notre modèle et y un vecteur contenant nos variables cible) ce qui nous donne le modèle de Logistic Regression :

$$\sigma(X.\theta) = \frac{1}{1+e^{-X.\theta}}$$

A partir de cette fonction, il est possible de définir une frontière de décision. Typiquement, on définit un seuil à 0.5 comme ceci :

$$\begin{cases} y = 0 \quad \text{si} \quad \sigma(X.\theta) < 0.5 \\ y = 1 \quad \text{si} \quad \sigma(X.\theta) \geq 0.5 \end{cases}$$

b) **Random Forest classifier (forêts d'arbres décisionnels)**
Random Forest est ce qu'on appelle une méthode d'ensemble (ou ensemble method en anglais) c'est-à-dire qu'elle "met ensemble" ou combine des résultats pour obtenir un super résultat finale. Random forest s'appuit sur les arbres de décicion, qui sont des outils qui aide à prendre

une décision grâce à une série de questions (aussi appelées tests) dont la réponse (oui/non) mènera à la décision finale. Les Random Forest peuvent être composées de plusieurs dizaines voire des centaines d'arbres, le nombre d'arbre est un paramètre que l'on ajuste généralement par "validation croisée" (ou cross-validation en anglais). Chaque arbre est entraîné sur un sous-ensemble du dataset et donne un résultat. Les résultats de tous les arbres de décision sont alors combinés pour donner une réponse finale. Chaque arbre "vote" (oui ou non) et la réponse finale est celle qui a eu la majorité de vote. Les étapes du Random forest peuvent être résumé comme ceci :

- On tire au hasard dans la base d'apprentissage B échantillons avec remise $z_i, i = 1, ...B$ (chaque échantillon ayant n points).
- Pour chaque échantillon i on construit un arbre de décision $G_i(x)$ selon un algorithme légèrement modifié : a chaque fois qu'un nœud doit être coupé (étape "split") on tire au hasard une partie des attributs(q parmis les p attributs) et on choisit le meilleur découpage dans ce sous-ensemble.
- Pour les problèmes de régression le résultat est obtenu sous la forme d'une agrégation par la moyenne $G(x) = \frac{1}{B}\sum_{i=1}^{B} G_i(x)$.
- Pour les problèmes de classification le résultat est obtenu sous la forme d'une agrégation par vote $G(x) = votemajoritaire(G_1(x), ..., G_B(x))$

c) **Gradient Boosting Machine (amplification de gradient)**
Le Gradient Boosting est un algorithme particulier de Boosting. Le Boosting consiste à assembler plusieurs « weak learners » pour en faire un « strong learner », c'est-à-dire assembler plusieurs algorithmes ayant une performance peu élevée pour en créer un beaucoup plus efficace et satisfaisant. Les algorithmes de Boosting se basent sur le même principe que ceux de Bagging. La différence apparaît lors de la création des « weak learner ». Pour le Boosting, les algorithmes ne sont plus indépendants au contraire, chaque «weak learner » est entraîné pour corriger les erreurs des « weak learner » précédents. L'assemblage de « weak learners » en « strong learner » se fait par l'appel successif de ceux-ci pour estimer une variable d'intérêt. Avec le gradient boosting dans le cadre d'une classification, chaque individu dispose d'un poids qui sera le même au départ, et qui, si un modèle se trompe, sera augmenté avant d'estimer le modèle suivant (qui prendra donc en compte ces poids), les weak learners utilisé pour l'algorithme de gradient sont généralement des arbres de décisions . Les modèles sont ajustés à l'aide d'une fonction de perte et d'un algorithme d'optimisation appelé descente de gradient [3]. Cela donne à la technique son nom, " gradient boosting ", car le gradient de perte est minimisé à mesure que le modèle est ajusté.

- **Principe du Boosting :**
  Pour les données d'apprentissage supervisé nous avons X_i nos features (variables prédictives) avec $X_i \in \mathbb{R}^d$ et Y_i nos labels (variable cible) avec $Y_i \in \mathbb{R}$ (Regression) ou $Y_i \in$ {0,1} (Classification).
  Considerons un ensemble de "weak learners" H, chaque learners h est un learners très simple et faible $h : \mathbb{R}^d \to \mathbb{R}$ ou $\mathbb{R}^d \to$ {0,1}

On combine additivement des weak learners pour obtenir des strong learners

$$g^{(B)}(X) = \sum_{b=1}^{B} \eta^{(b)} h^{(b)}(X)$$

avec $\eta^{(b)} \geq 0$ pour espérer en obtenir un meilleur. Chaque $b = 1, ...B$ est un pas/itération de boosting.

- **Principe du gradient boosting :**
  On cherche donc des fonctions $h^{(b)}$ du dictionnaire H et des réel $\eta^{(b)}$ tels que

$$g^{(B)}(X) = \sum_{b=1}^{B} \eta^{(b)} h^{(b)}(X)$$

minimise le risque empirique

$$\sum_{i=1}^{n} L(Y_i, g^{(b)} X_i),$$

où L est la fonction de coût (de perte) que l'on se fixe suivant le problème.

2.2.5.1 Évaluation des modèles construits

Les modèles de machines learning construit sur la base des algorithmes abordé précédement on ensuite été évaluer sur l'ensemble de test, les résutats obtenus sont résumés dans le tableau 2.3 suivant :

	ACCURACY	PRECISION	RECALL	F1 score	AUC
Logistique regression	0.76	0.28	0.80	0.41	0.778
Random forest classifier	0.91	0.61	0.47	0.53	0.717
Gradient Boosting Machine	0.89	0.48	0.60	0.54	0.761

TABLE 2.3 : Évaluation des modèles construits

Après avoir comparé les performances de tout les modèles construis, le modèle avec l'algorithme Gradient Boosting Machine (GBM) est le meilleur suivi ensuite du modèle avec Random Forest classifier et enfin du modèle construit avec l'algorithme de Logistique regression.
En effet une chose importante à noter ici est que nous voulons un score de rappel (RECALL) élevé même s'il conduit à un score de précision faible (conformément au compromis precision-recall que nous recherchons). Nous nous soucions davantage de minimiser les faux négatifs, c'est-à-dire les personnes qui ont été prédites comme non-défaillants par le modèle mais qui étaient en réalité des défaillants. Nous ne voulons pas manquer un défaillant comme étant classé comme non défaillant car le coût des erreurs pourrait être très élevé. Bien que des trois modèles celui de la regression logistique obtient le score de recall le plus élevé (0.80), son score de précision reste très bas (0.28) et ne nous permet pas d'atteindre le compromis precision-recall que nous recherchons. Le modèle de Random forest quant à lui obtient le meilleur score de justesse mais son équilibre entre les scores de précision et de recall n'est pas le meilleur après évaluation de l'ensemble des modèles construit. Le modèle construit avec l'algorithme de Gradient Boosting Machine (GBM) offre le meilleur compromis precision-recall que nous recherchons, avec un score de precision de 0.48, un score de recall de

0.60, un F1-score de 0.54 et un score de justesse (Accuracy) de 0.89 , la valeur de son AUC supérieure à 0.70 témoigne également de sa bonne capacité dans la discrimination des classes négative et positive.

2.2.5.2 Optimisation du modèle sélectionné

Le modèle de gradient boosting sélectionné à l'étape précédente a ensuite été amélioré grâce à l'optimisation de certains de ses Hyperparamètres, un processus qui a consisté en la recherche de la configuration des hyperparamètres qui produit les meilleures performances, nous avons utiliser la méthode GridSearchCV de Sklearn paramétrer avec la métrique AUC pour éffectuer ce processus. Afin d'éviter un traitement long et coûteux en terme de calcul juste trois hyperparamètres ont été optimisés. Les valeurs d'hyperparamètres obtenues sont les suivantes :

{'gradientboostingclassifier__learning_rate' : 0.05,
'gradientboostingclassifier__max_depth' : 3,
'gradientboostingclassifier__n_estimators' : 100}

- **Le learning_rate** détermine l'impact de chaque arbre sur le résultat final. GBM fonctionne en commençant par une estimation initiale qui est mise à jour à l'aide de la sortie de chaque arbre. Le paramètre d'apprentissage contrôle l'ampleur de ce changement dans les estimations.
- **max_depth** représente la profondeur maximale d'un arbre, il est utilisé pour contrôler le sur-ajustement car une profondeur plus élevée permettra au modèle d'apprendre des relations très spécifiques à un échantillon particulier, ce paramètre est mieux régler avec une Cross-Validation.
- **n_estimators** représente le nombre d'arbres séquentiels à modéliser, bien que GBM soit assez robuste à un nombre plus élevé d'arbres, il peut toujours sur-adapter (sur-ajuster) à un moment donné, ce nombre est donc choisi à l'aide d'une cross-validation.

Le modèle final obtenu sera ensuite validé, une étape présenté dans le chapitre suivant de notre mémoire

2.2.6 Déploiement du modèle à travers une API

Le modèle obtenu après la phase de modélisation a été enregistré sur le disque à l'aide du modèle de persistance intégré de Python (pickle) et ensuite déployé à travers une API construit avec le micro framework Flask[2], cette API via une requête POST reçoit les données sur un client et renvoie en temps réel le résultat de la prédiction du modèle. Les données entrées sont contenues dans un tableau à une dimention et treize colonnes (1,13) appelé "feature_array", ce sont les valeurs des variables prédictives pour un client, le résultat obtenu est renvoyé dans un format Json, ce résultat se compose de la probabilité de remboursement, de la probabilité de non rembourcement et du statut solvable ou pas du client.

Conclusion

Nous avons présenté dans ce chapitre les différents outils et matériels physiques comme logiciels ainsi que les technologies et la base de données qui nous ont permis d'effectuer nos travaux. Nous avons également décrit les différentes phases nécessaires à la mise en place de la solution. Dans le chapitre suivant sera présenté,les résultats de notre modèle finale et une discussion sur ses perfomances.

Chapitre 3

Résultats et discussion

Les chapitres précédents nous ont permis de faire un état de l'art et de présenter les outils et méthodes dont nous avons fait usage pour l'élaboration de notre solution. Dans ce chapitre, nous présentons les résultats obtenus en se basant sur les précédents chapitres de ce document.

3.1 Évaluation du modèle finale

Après optimisation de certains hyperparamètres de notre modèle basé sur le gradient boosting machine, abordé dans le chapitre précédent, le nouveau modèle obtenu a été évalué sur notre ensemble de test :

a) **Métriques d'évaluation**
Les scores obtenu par notre modèle final est résumé dans le tableau 3.1 suivant :

ACCURACY	PRECISION	RECALL	F1 score	AUC
0.88	0.45	0.65	0.53	0.776

TABLE 3.1 : Évaluation de notre modèle finale

On peut remarquer une amélioration du recall et de l'AUC qui sont passés de 0.60 pour le recall à 0.65 et de 0.761 pour l'AUC à 0.776 entre le modèle de GBM initial et notre modèle final, on remarque également une légère diminution de l'accuracy et de la précision qui sont passés respectivement de 0.89 à 0.88 et de 0.48 à 0.45 entre nos deux modèles.

b) **Graphique de gain cumulé et de levage(Gain and Lift charts)**
Les graphiques Gain et Lift sont utilisés pour évaluer les performances des modèles de classification, ils messurent les avantages de l'utilisation d'un modèle et sont utilisés dans des contextes commerciaux tels la modélisation des risques, le marketing ciblé, etc.
Nous avons utilisé les graphiques de gain et de levage et avons obtenus les resultats ci-après :

- Graphique de gain cumulé (Cumulative gain curve)

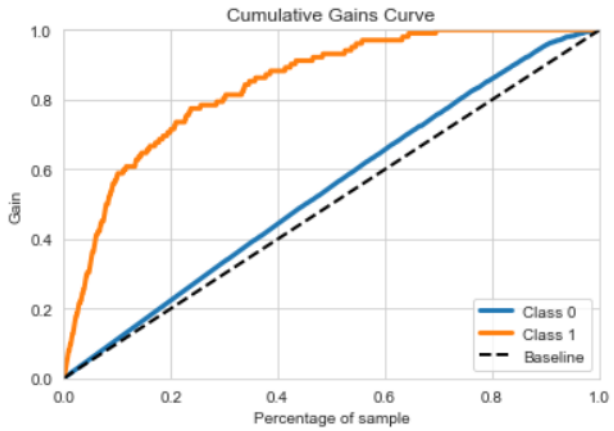

FIGURE 3.1 : Graphique de gain cumulé

Le graphique de gain ci-dessus (figure 2) montre que Plus de 75 % mais moins de 80% des défaillants sont capturés dans les 20% supérieur des données sur la base de notre modèle.

- Graphique de levage (Lift curve)

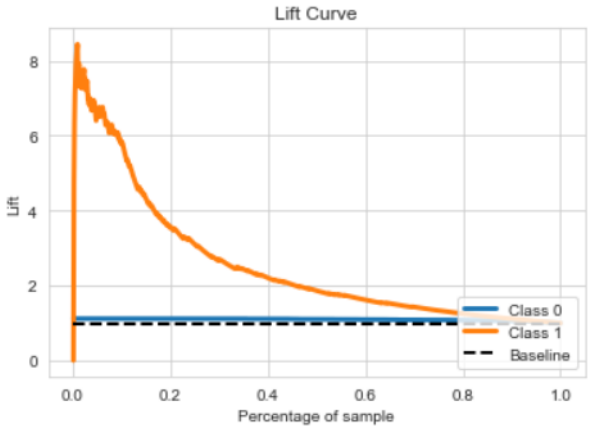

FIGURE 3.2 : Graphique de levage

Lift Curve indique que plus la valeur de lift cumulée est élevée, meilleure est la précision (accuracy). On observe dans notre cas à partir de notre graphique de lift (figure 2) une augmentation du lift cumulée d'environs 3,90 pour les 2 déciles supérieurs de nos données, ce qui signifie que avec les prédictions de notre modèle sur les 20% supérieur de nos données, on peut s'attendre à 3,90 fois le nombre total de défaillants trouvé sur la base d'une prédiction sans modèle, sur les mêmes 20% supérieur de nos données.

c) **Importance des fonctionnalités (Features importance)**

L'importance des fonctionnalités permet de comprendre l'utilité des fonctionnalités (variables prédictives d'entrée) dans la prédiction de la variable cible par notre modèle. Dans notre cas, l'importance fournit un score qui indique l'utilité ou la valeur de chaque caractéristique dans la construction des arbres de décision améliorés au sein du modèle de gradient bossting.

L'utilité de nos fonctionnalités d'entrée dans la prédiction de la variable cible par notre modèle finale de gradient bossting est résumé par le graphique (figure 3) suivant :

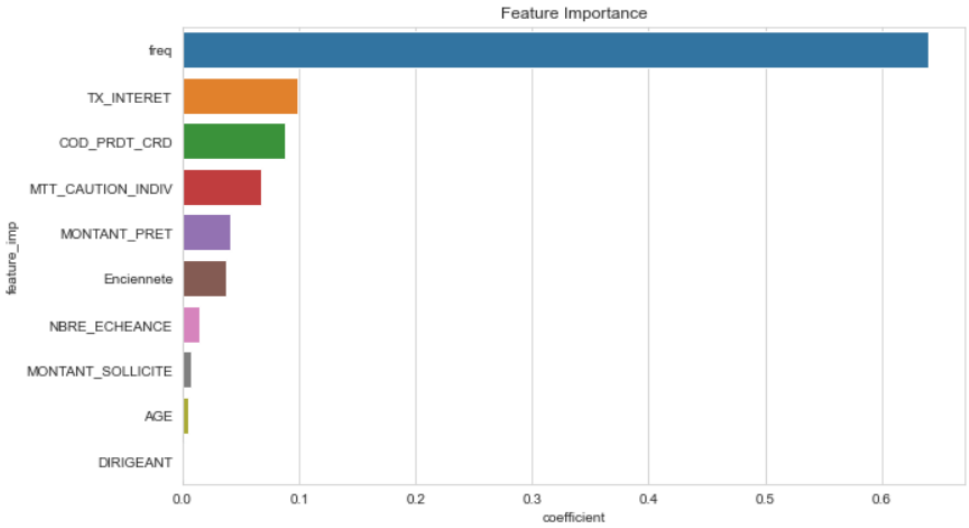

FIGURE 3.3 : Importance des fonctionnalités d'entrées

On peut remarquer à travers ce graphique que c'est la variable "freq"(fréquence d'emprunt) qui est largement la plus utile dans le processus de prédiction de la variable cible, elle est suivie des autres fonctionnalités dont les variables "TX_INTERET" (Taux d'intérêt) et "COD_PRDT_CRDT" (produit de crédit) dont les utilités sont également non négligeables.

3.1.1 Test réalisé sur notre API

Comme nous l'avons dit dans le chapitre précédent, notre API via une requête POST reçoit les données sur un client dans un tableau à une dimention et treize colonnes(1,13) appelé "feature_array" et renvoie en temps réell le résultats de la prédiction du modèle sous un format Json, nous avons eu à tester notre API avec l'outils de test d'API postman (figure 3.1.1 et 3.1.1) :

- Prédiction d'un client comme défaillant

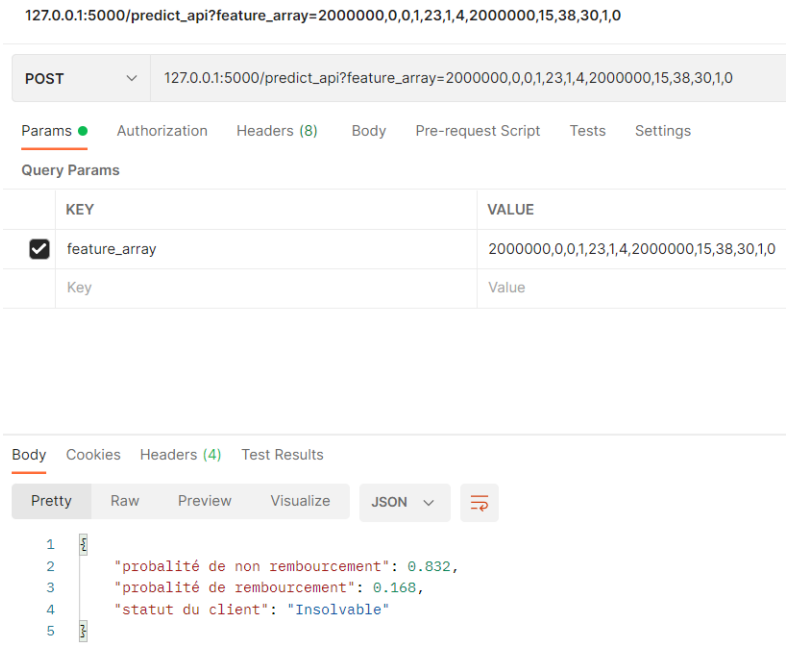

FIGURE 3.4 : Réponse de l'API pour un client prédit comme défaillant

- Prédiction d'un client comme non défaillant

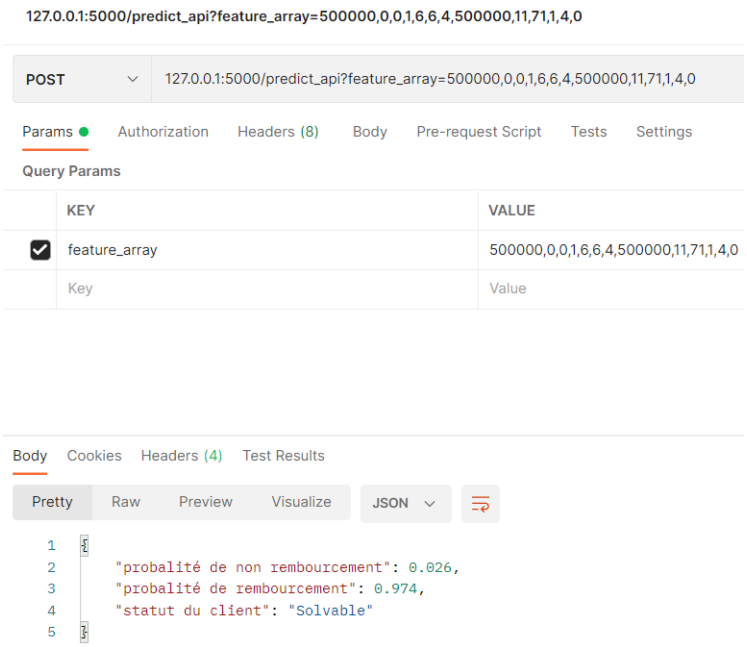

FIGURE 3.5 : Réponse de l'API pour un client prédit comme non défaillant

3.2 Discussion

Au vu des résultats obtenus nous pouvons dire que les objectifs visés ont été atteints. En effet nous avons pu créer un système de modélisation du risque de crédit pour prédire les probabilités de défaut d'un emprunteur en utilisant les données historiques sur les crédits individuels effectuer au sein de la MDB. Nous avons effectué une collecte des données, réalisé une analyse exploratoire de ces données, construis plusieurs modèles de machine learning et sur la base des résultats obtenue après évaluation de ces modèles nous avons pu dégager le modèle le plus optimal et le mieux adapté pour la résolution de notre problème, ce modèle a ensuite été amélioré et intégreé dans une API pour son déploiement. Le modèle final obtenu à la suite de notre travail a eu un score de rappel (recall) de 0.65 ce qui traduit une assez bonne identification des clients défaillants et un score de précion (precision) de 0.48 sur notre ensemble de données de test, ce qui traduit une identification correcte des personnes non défaillantes assez limitée. Bien que notre objectif était de maximiser le rappel au détriment de la précision, une précision de 0.48 est assez faible car l'identification correcte des personnes non défaillantes est également très importantes, l'idéal serait donc d'atteindre un compromis précision recall avec les meilleures valeurs possible de ces deux métriques.
L'analyse du graphique de levage (Lift Curve) nous indique également que avec les prédictions de notre modèle sur les 20% supérieur de nos données, on peut s'attendre à 3.90 fois le nombre total de défaillant trouvé sur la base d'une prédiction sans notre modèle, sur les mêmes 20% supérieures de nos données, cela montre de l'avantage que notre modèle pourrait apporter en tant que aide à la décision dans l'évaluation du risque de crédit au sein de la MDB.

Conclusion

Dans ce chapitre, nous avons présenté les résultats de l'évaluation de notre modèle, le résultats du test de notre API qui a servi à déployer ce modèle ainsi que les insuffisances de notre travail. Sur la base de ces insuffisances nous formulerons des perspectives pour améliorer notre solution.

Conclusion Générale

Dans ce travail nous avons proposé un système capable de modéliser le risque de crédit pour une institution de microfinance dénommée MDB et ainsi prédire les probabilités de défaut d'un emprunteur, en utilisant les données historiques sur les crédits individuels. Notre objectif a été atteint à travers la réalisation d'un modèle d'apprentissage automatique basé sur l'algorithme de gradient bossting machine et la réalisation d'une API pour le déploiement de notre modèle. Cependant le travail réalisé pourrait être amélioré. Notre système pourrait être plus performant avec l'utilisation d'un ensemble de données plus vaste, une stratégie plus élaborée dans l'ingénieurie des fonctionnalités et l'utilisation de techniques plus perfommantes tel que l'apprentissage en profondeur avec les réseaux de neurones artificiels. Une autre perspective pertinante serait de mettre en place une stratégie orientée MLOps pour une surveillance et un réentraînement continuels du modèle en production afin de s'assurer qu'ils fonctionnent de manière optimale à mesure que les données changent au fil du temps.

Bibliographie

[1] Apostolos Ampountolas, Titus Nyarko Nde, Paresh Date, and Corina Constantinescu. A machine learning approach for micro-credit scoring. 2021.

[2] Salim Lardjane. Python et technologies web.

[3] Ricco RAKOTOMALALA. Gradient boosting : Technique ensembliste pour l'analyse prédictive, introduction explicite d'une fonction de coût.

[4] Guillaume Saint-Cirgue. Apprendre le machine learning en une semaine.

[5] CHIBEL ZINEB, BAMOUSSE Zineb, and EL KABBOURI Mounime. Revue du contrôle de la comptabilité et de l'audit : Etude de différentes méthodes d'analyse de risque crédit : Revue de littérature-numéro 7. 2018.

Webographie

[6] Rafael Bastos. Credit risk analysis with machine learning. https://towardsdatascience.com/credit-risk-analysis-with-machine-learning-736e87e95996. Consulté le 20/01/2022.

[7] Brain cube. Ia vs. machine learning vs. science des données (data science) pour l'industrie. https://fr.braincube.com/resource/ia-par-rapport-au-machine-learning-par-rapport-a-la-science-des-donnees-pour-le-secteur/. Consulté le 03/01/2021.

[8] DevanshiKulshreshtha. loan_data_2007_2014. https://www.kaggle.com/devanshi23/loan-data-2007-2014. Consulté le 21/02/2022.

[9] Bastien L. Machine learning : Définition, fonctionnement, utilisations. https://datascientest.com/machine-learning-tout-savoir. Consulté le 06/01/2022.

[10] Asad Mumtaz. How to develop a credit risk model and scorecard. https://towardsdatascience.com/how-to-develop-a-credit-risk-model-and-scorecard-91335fc01f03. Consulté le 20/01/2022.

[11] Oracle. Qu'est-ce que la data science ?

[12] Underwrite.ai. Underwrite.ai. https://www.underwrite.ai/. Consulté le 20/01/2022.

[13] Kylle Wiggers. Zest raises $15 million to reduce loan algorithm bias. https://venturebeat.com/2020/10/20/zest-raises-15-million-to-reduce-loan-algorithm-bias/. Consulté le 20/01/2022.

[14] Wikipedia. Apprentissage automatique. https://fr.wikipedia.org/wiki/Apprentissage_automatique. Consulté le 06/01/2022.

[15] Wikipedia. Risque de crédit. https://fr.wikipedia.org/wiki/Risque_de_cr%C3%A9dit. Consulté le 06/01/2022.

Table des matières