

PROJECT 7 - PARCOURS DATA SCIENTIST « IMPLEMENTEZ UN MODELE DE SCORING

**Note Méthodologique**



23 février 2023

Mohamed Assali

Table des matières

[2 Implémentez un modèle de scoring : Note méthodologique 1](#_Toc128085691)

[2.1 Contexte : 1](#_Toc128085692)

[2.2 Prétraitement de données 2](#_Toc128085693)

[2.2.1 Valeurs aberrantes 2](#_Toc128085694)

[2.2.2 Feature Enginering 2](#_Toc128085695)

[2.2.3 Automatique Feature Engineering 3](#_Toc128085696)

[2.2.4 Création manuelle de nouvelles Features 3](#_Toc128085697)

[2.2.5 Valeurs manquantes 4](#_Toc128085698)

[2.2.6 Catégorisation des variables catégorielles 4](#_Toc128085699)

[2.3 Selection d’un modèle de Machine Learning et méthode d’entrainement 4](#_Toc128085700)

[2.3.1 Machine Learning : C’est quoi ? 5](#_Toc128085701)

[2.4 Démarche de modélisation 5](#_Toc128085702)

[2.4.1 Équilibrage des classes de la variable cible 5](#_Toc128085703)

[2.4.2 Optimisation des paramètres de modèle 7](#_Toc128085704)

[2.4.3 Fonction coût : métrique bancaire 8](#_Toc128085705)

[2.4.4 Seuil de solvabilité 9](#_Toc128085706)

[2.4.5 Interprabilité du modèle 10](#_Toc128085707)

[2.4.6 Limites et améliorations possibles 12](#_Toc128085708)

# Implémentez un modèle de scoring : Note méthodologique

## Contexte :

Le scoring de crédit est un outil important pour les institutions financières, qui leur permet de mesurer la probabilité qu'un client rembourse son crédit et ainsi de prendre des décisions éclairées en matière d'octroi de crédit. Dans ce contexte, l'entreprise Prêt à dépenser a décidé de développer un algorithme de classification basé sur des sources de données variées, afin de mieux évaluer la solvabilité de ses clients.

Prêt à dépenser a également pris en compte la demande croissante de transparence de la part de ses clients. Ainsi, elle a décidé de développer un dashboard interactif qui permettra aux chargés de relation client d'expliquer de façon transparente les décisions d'octroi de crédit, tout en permettant aux clients de consulter leurs informations personnelles et de les explorer facilement.

Le rapport fournit une explication détaillée des différentes étapes de modélisation qui ont été nécessaires pour mener à bien le projet.

## Prétraitement de données

Le prétraitement des données est une étape essentielle dans le processus de modélisation de scoring de crédit. Cette étape consiste à nettoyer le jeu de données en supprimant les valeurs aberrantes et les variables avec des valeurs manquantes excessives. Ensuite, les variables catégorielles doivent être numérisées pour rendre les données interprétables pour les algorithmes de machine learning.

Cette étape peut également inclure du "feature engineering", c'est-à-dire la création de nouvelles variables à partir des variables existantes pour améliorer la performance du modèle. Le prétraitement des données peut être un processus complexe et fastidieux, mais il est essentiel pour garantir la qualité des données utilisées dans le processus de modélisation.

### Valeurs aberrantes

Il est important de traiter les valeurs aberrantes dans le jeu de données d'entraînement et de test, car elles peuvent fausser les résultats du modèle de scoring de crédit. Les valeurs rares sont des valeurs qui ont une fréquence très faible dans le jeu de données, tandis que les valeurs aberrantes sont des valeurs qui se situent en dehors de l'intervalle attendu pour une variable donnée.

Dans le jeu de données d'entraînement, les valeurs rares ou aberrantes doivent être identifiées et traitées en utilisant des techniques appropriées telles que la suppression de lignes ou la substitution par une valeur plus plausible. Les mêmes techniques doivent être appliquées au jeu de données de test pour garantir que le modèle est capable de généraliser aux données inconnues.

Dans le cadre de ce projet, certaines variables catégorielles ont été examinées pour détecter des valeurs rares, et il a été constaté que la valeur "XNA" n'apparaissait que 4 fois dans la variable "CODE GENDER". Ces valeurs rares ont été traitées en les fusionnant avec d'autres catégories similaires ou en les supprimant si elles n'étaient pas suffisamment significatives.

Pour les variables numériques, des valeurs aberrantes ont été détectées, notamment pour la variable "DAYS\_EMPLOYED" qui comportait des valeurs supérieures à 366 jours. Les emprunts correspondant à ces valeurs ont été supprimés du jeu de données pour éviter d'affecter négativement les résultats du modèle

### Feature Enginering

### Automatique Feature Engineering

Afin d'améliorer la qualité des données et de fournir des informations supplémentaires pour le modèle de scoring de crédit, d'autres jeux de données contenant des informations complémentaires ont été agrégés aux ensembles d'entraînement et de test. Cette opération a pour but d’améliorer la qualité des données et à fournir au modèle un contexte plus complet pour l'évaluation des demandes de crédit.

La technique d'agrégation est un processus important pour obtenir des informations supplémentaires à partir des données brutes. Dans cette technique, les fonctions qui ont été utilisées sont la somme (sum), la moyenne (mean) et la somme cumulée (cumsum)….

La somme est utilisée pour agréger les valeurs numériques, permettant de calculer la somme totale pour chaque variable d'intérêt.

La moyenne, quant à elle, est utilisée pour obtenir une mesure de tendance centrale pour les variables numériques. En utilisant la moyenne, on peut obtenir une estimation de la valeur typique de chaque variable, ce qui peut être utile pour détecter les valeurs aberrantes ou extrêmes.

La somme cumulée, quant à elle, est utilisée pour calculer la somme des valeurs pour chaque variable, en partant de la première observation jusqu'à la dernière.

Cette méthode peut être utilisée pour suivre l'évolution des données sur une période de temps, comme le remboursement d'un prêt. En somme, la technique d'agrégation est un outil clé dans la modélisation de scoring de crédit, qui permet d'obtenir des informations supplémentaires à partir des données brutes et ainsi de mieux comprendre les comportements des clients en matière de remboursement de crédit.

### Création manuelle de nouvelles Features

Les variables additionnelles sont des caractéristiques qui ont été créées manuellement à partir des variables originales du jeu de données. Ces variables ont été conçues pour mieux capturer les relations potentielles entre les variables d'origine et la variable cible (la probabilité de remboursement du prêt).

Dans le cadre de ce projet, quelques variables additionnelles ont été construites, telles que la proportion du temps travaillé par rapport à l’âge du client, qui permet de mesurer le rapport entre le temps de travail et l'âge du client, ou encore le ratio crédit/revenu, qui représente la proportion du revenu mensuel du client qui est consacré au remboursement de ses crédits. Ces variables ont été intégrées aux jeux de données d'entrainement et de test pour améliorer la performance du modèle de scoring.

### Valeurs manquantes

Certaines variables peuvent contenir des données manquantes, ce qui peut poser un problème pour les modèles de machine learning qui nécessitent des valeurs numériques pour chaque variable. Bien qu'il existe plusieurs méthodes pour remplacer les valeurs manquantes, il est préférable que leur nombre ne soit pas trop élevé. Ainsi, dans le cadre de ce projet, toutes les variables présentant un taux de données manquantes supérieur à 30% ont été supprimées du jeu de données d'entrainement et de test. Pour les autres variables, les valeurs manquantes ont été remplacées par la valeur médiane de la variable.

### Catégorisation des variables catégorielles

Lorsque les données sont issues de sources variées, il est courant de rencontrer des variables catégorielles qui ne peuvent être interprétées directement par les modèles de ML. Pour pouvoir les utiliser, il est nécessaire de les encoder en variables numériques. Plusieurs techniques d'encodage peuvent être utilisées, et dans le cadre de ce projet la technique *One-Hot Encoding* a été utilisée. L’image ci-dessous montre un tableau explicatif du principe de fonctionnement de la technique *ne-Hot Encoding.*



Figure 1: Tableau explicatif de la technique ne-Hot Encoding

Cette technique consiste à créer une nouvelle colonne pour chaque valeur unique de la variable catégorielle, où la colonne correspondante sera remplie de 1 si la valeur de la variable est égale à cette valeur unique, et de 0 sinon.

La partie suivante d’écrit la procédure de sélection d’un modèle de classification.

## Selection d’un modèle de Machine Learning et méthode d’entrainement

### Machine Learning : C’est quoi ?

Le machine learning est une méthode d'analyse de données qui permet aux ordinateurs d'apprendre et d'améliorer leurs performances sans être explicitement programmés. Cette technique est de plus en plus utilisée dans de nombreux domaines, y compris la finance, le commerce électronique, la santé et la sécurité, pour résoudre des problèmes complexes tels que la détection de fraudes, la reconnaissance de motifs et la classification de données.

Le machine learning est fondé sur des algorithmes qui s'entraînent sur des données et apprennent à identifier des motifs ou des règles à partir de ces données. Ces algorithmes peuvent être supervisés ou non supervisés, selon qu'ils disposent ou non d'un ensemble de données étiquetées pour l'apprentissage. Les modèles de machine learning peuvent être utilisés pour effectuer des prévisions et prendre des décisions automatisées.

Des exemples de techniques de machine learning incluent la régression linéaire, les arbres de décision, les réseaux de neurones, les machines à vecteurs de support (SVM) et les algorithmes de forêt aléatoire. Des bibliothèques logicielles populaires pour le machine learning incluent TensorFlow, Keras, scikit-learn, PyTorch et Apache Spark.

Le machine learning a révolutionné de nombreux domaines et a permis d'obtenir des résultats impressionnants, notamment dans la reconnaissance d'image, la reconnaissance vocale et la traduction automatique. Cependant, il est important de noter que le machine learning n'est pas une solution miracle et qu'il peut être sujet à des biais et des erreurs. Il est donc important de mettre en place des pratiques de vérification et de validation pour garantir la qualité et la fiabilité des résultats obtenus.

Parmi les modèles couramment utilisés, nous avons choisi les modèles suivants : Dummy, Forêts aléatoires, et LGBM.

Nous avons effectué des tests sur tous ces modèles avec leurs paramètres par défaut pour avoir une première idée de leur comportement sur nos données. Les résultats ont montré LGBM est le modèle le plus performant, nous avons donc sélectionné ce modèle pour l’optimisation de ses paramètres

## Démarche de modélisation

### Équilibrage des classes de la variable cible

Si nous examinons de près la variable "TARGET", nous pouvons constater qu'il y a un déséquilibre important dans les données. Cela signifie qu'il y a une disparité significative entre le nombre d'observations appartenant à chaque catégorie de la variable "TARGET"

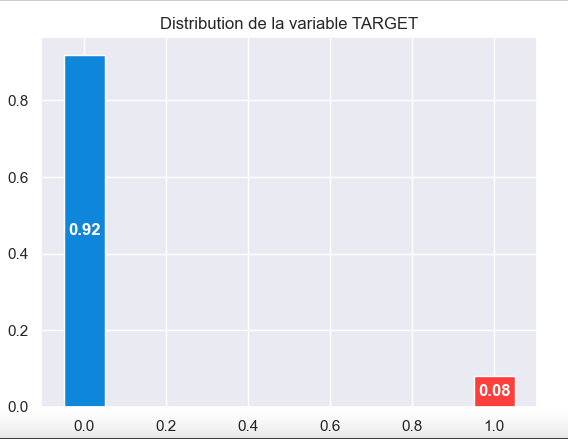


Figure 2: Distribution de la variable 'TARGET'

Dans le but d'équilibrer notre ensemble de données d'entraînement, nous avons commencé par utiliser des méthodes de suréchantillonnage. Cette approche nous a permis de générer des données synthétiques à l'aide de la librairie SMOTE, ce qui a permis d'obtenir un équilibre satisfaisant entre les différentes classes de notre jeu de données et ainsi améliorer la qualité de l'apprentissage.

SMOTE est une technique de suréchantillonnage (Synthetic Minority Over-sampling Technique) qui permet de générer des données synthétiques pour les classes minoritaires dans un ensemble de données déséquilibré. Cette technique consiste à créer de nouvelles observations pour la classe minoritaire en interpolant les caractéristiques de voisins proches.

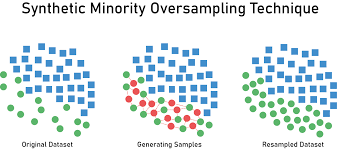


Figure 3: Image explicatif du principe de fonctionnemet de la méthode SMOTE

En d'autres termes, elle permet de créer de nouveaux exemples synthétiques pour les classes minoritaires en utilisant une combinaison linéaire des caractéristiques de leurs voisins proches. Cela permet d'équilibrer l'ensemble de données et d'améliorer la qualité de l'apprentissage en évitant les biais induits par des classes sous-représentées.

### Optimisation des paramètres de modèle

Comme mentionné précédemment, chaque modèle de Machine Learning requiert la spécification de différents paramètres. En fonction de la combinaison de ces paramètres choisie pour le modèle, les performances obtenues pourront varier, allant d'une grande efficacité à une moins bonne qualité des résultats.

La méthode GridSearchCV est une approche courante pour trouver les meilleurs paramètres pour un modèle de Machine Learning.

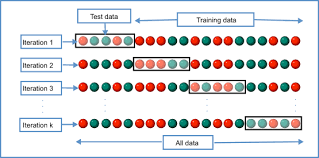


Figure 4: Image explicatif du principe de fonctionnement de la méthode GridSearch CV

Elle consiste à fixer un ensemble de valeurs pour chaque paramètre du modèle, puis à évaluer les performances du modèle en utilisant ces différentes combinaisons de paramètres. Cette méthode permet de passer en revue toutes les combinaisons possibles de paramètres, ce qui peut être fastidieux à faire manuellement. Pour évaluer les performances, on utilise une procédure de validation croisée (cross-validation), qui consiste à diviser les données en plusieurs ensembles (ou "folds"), pour ensuite entraîner le modèle sur une partie de ces ensembles et le tester sur les autres. En répétant cette procédure plusieurs fois, on peut obtenir une estimation plus robuste des performances du modèle pour chaque combinaison de paramètres. Finalement, la méthode GridSearchCV retourne les paramètres qui ont produit les meilleures performances.

### Fonction coût : métrique bancaire

Lorsqu'on compare plusieurs modèles de Machine Learning, on peut les évaluer en fonction de leurs performances, mesurées selon une métrique particulière. Cette métrique dépendra du type de problème et des objectifs poursuivis, mais elle peut être par exemple la précision (accuracy), le rappel (recall) ou le F1-score. En général, on cherchera à maximiser cette métrique, ce qui correspond à obtenir les meilleurs résultats possibles pour le problème considéré.

Dans le cadre de ce projet, une fonction de coût spécifique a été développée pour évaluer les performances des modèles. Cette fonction de coût a pour objectif de maximiser les gains obtenus par la validation du modèle. Les gains sont mesurés en termes de bénéfices et de coûts, par exemple le gain financier réalisé ou le temps économisé. En optimisant cette fonction de coût, on cherchera à trouver les paramètres du modèle qui maximisent les gains obtenus par la validation, tout en minimisant les pertes.

Ainsi, le modèle qui permettra d'obtenir les meilleurs résultats selon cette fonction de coût sera considéré comme étant le plus performant pour le problème considéré. Cette approche permet de prendre en compte les contraintes et les objectifs spécifiques au problème étudié, pour choisir le modèle qui conviendra le mieux à la tâche à accomplir.

Il s'agit d'un problème de classification binaire, où chaque exemple possède une étiquette qui peut prendre l'une des deux valeurs possibles : 0 ou 1. L'étiquette 0 correspond à un exemple qui est considéré comme négatif, c'est-à-dire solvable, alors que l'étiquette 1 correspond à un exemple positif, qui est considéré comme non solvable.

Lorsqu'on évalue un modèle de classification binaire, on peut comparer ses prédictions aux valeurs réelles des exemples. Cela permet de déterminer s'il prédit correctement la classe de chaque exemple ou non. Dans ce contexte, il existe quatre combinaisons possibles :

Les vrais négatifs (True Negatives, TN) : le modèle prédit la classe 0 et la valeur réelle est également 0.

Les vrais positifs (True Positives, TP) : le modèle prédit la classe 1 et la valeur réelle est également 1.

Les faux négatifs (False Negatives, FN) : le modèle prédit la classe 0 alors que la valeur réelle est 1.

Les faux positifs (False Positives, FP) : le modèle prédit la classe 1 alors que la valeur réelle est 0.

Ces quatre combinaisons permettent de déterminer la qualité des prédictions du modèle en termes de sensibilité (capacité à détecter les vrais positifs), de spécificité (capacité à détecter les vrais négatifs), de précision (rapport entre les vrais positifs et tous les exemples prédits comme positifs) et de rappel (rapport entre les vrais positifs et tous les exemples réellement positifs).

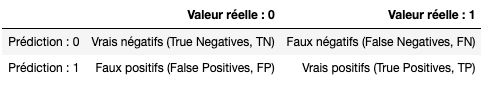


Figure 5: Table de confusion pour la classification binaire

Ainsi, nous avons défini la fonction de coût J pour évaluer la qualité de notre modèle. Cette fonction se base sur les valeurs de vrais positifs (TP), de vrais négatifs (TN), de faux positifs (FP) et de faux négatifs (FN), et utilise des coefficients pondérés pour chacun de ces quatre types de prédictions.

|  |  |  |
| --- | --- | --- |
|  |  | *Équation 1* |
|  |  |  |

Les coefficients pondérés ont été choisis arbitrairement pour ce projet, avec des valeurs spécifiques pour chaque type de prédiction. En particulier, TP\_value est égal à 0, FN\_value est égal à -10, TN\_value est égal à 1 et FP\_value est égal à 0.

Ces valeurs de coefficients reflètent le fait que les faux négatifs ont des conséquences plus graves que les vrais négatifs, tandis que les vrais positifs et les faux positifs ont un impact moindre sur la fonction de coût. Par exemple, un faux négatif entraîne une perte de 10, tandis qu'un vrai négatif n'a pas d'impact sur la fonction de coût (TN\_value = 1). De même, un vrai positif ou un faux positif ont tous deux un impact nul sur la fonction de coût.

La fonction de coût J permet donc de quantifier les coûts et les avantages associés aux différents types de prédictions. En maximisant la valeur de J, on peut trouver les paramètres optimaux pour le modèle de classification binaire.

### Seuil de solvabilité

Le modèle de classification binaire retourne un score compris entre 0 et 1, où une valeur proche de 1 indique une forte probabilité que le client soit non solvable, et une valeur proche de 0 indique une forte probabilité qu'il soit solvable. Par défaut, le modèle utilise un seuil de 0,5 pour prendre la décision finale de classification : si le score prédit est supérieur à 0,5, le client est considéré comme non solvable, sinon il est considéré comme solvable.

Cependant, il est possible d'optimiser le seuil de classification pour améliorer les performances du modèle. Dans ce projet, le seuil optimal a été déterminé à l'aide de la méthode "hyperopt", qui permet d'optimiser les hyperparamètres d'un modèle en minimisant une fonction de coût spécifiée.

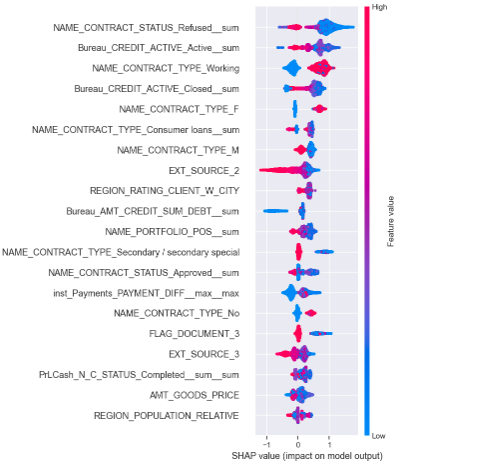
En ajustant le seuil de classification, on peut réduire le nombre de faux positifs ou de faux négatifs du modèle, en fonction des objectifs spécifiques de l'application. Par exemple, si l'on souhaite minimiser le nombre de clients classés à tort comme non solvables, on peut augmenter le seuil de classification, ce qui réduira le taux de faux positifs mais augmentera le taux de faux négatifs. À l'inverse, si l'on souhaite minimiser le nombre de clients classés à tort comme solvables, on peut diminuer le seuil de classification, ce qui réduira le taux de faux négatifs mais augmentera le taux de faux positifs.

L'optimisation du seuil de classification permet donc d'ajuster le modèle en fonction des besoins spécifiques de l'application, et peut conduire à une amélioration significative des performances du modèle de classification binaire.

### Interprétabilité du modèle

Le modèle étant destiné à des équipés opérationnelles devant́ être en mesure d’expliquer les décisions de l’algorithme à des clients réels, le modèle est accompagné́ d’un module d’explicabilité.

Pour réaliser ce module, la première perspective envisagée était d’utiliser l’importance des features issues de modèle utilisé .



L'approche que nous avons adoptée a consisté à utiliser shap.Explainer, qui est un outil permettant d'interpréter les prédictions des modèles de machine learning. Cette approche permet d’afficher les valeurs de Shaplet de chaque feature.

Les valeurs de Shapley (également appelées shaplets) sont un moyen d'expliquer l'importance de chaque caractéristique dans la prédiction d'un modèle de machine learning pour un échantillon de données spécifique. Les valeurs de Shapley sont calculées en utilisant la théorie des jeux coopératifs, qui considère chaque caractéristique comme un joueur dans un jeu et mesure leur contribution individuelle à la prédiction du modèle. Les valeurs de Shapley sont ensuite utilisées pour attribuer une importance relative à chaque caractéristique dans la prédiction du modèle pour cet échantillon de données spécifique. En utilisant les valeurs de Shapley, les équipes opérationnelles peuvent expliquer pourquoi une prédiction a été faite pour un client spécifique en mettant en évidence les caractéristiques les plus influentes dans la prise de décision.

Lorsque la valeur SHAP pour une caractéristique est proche de 1, cela indique que cette caractéristique a une forte influence positive sur la prédiction. Cela signifie que des valeurs élevées de cette caractéristique conduisent à des valeurs de sortie plus élevées du modèle.

Inversement, lorsque la valeur SHAP pour une caractéristique est proche de -1, cela indique que cette caractéristique a une forte influence négative sur la prédiction. Cela signifie que des valeurs élevées de cette caractéristique conduisent à des valeurs de sortie plus faibles du modèle.

Enfin, lorsque la valeur SHAP pour une caractéristique est proche de 0, cela indique que cette caractéristique a peu ou pas d'influence sur la prédiction.

Les couleurs utilisées dans le graphique de résumé SHAP indiquent simplement la valeur de SHAP pour chaque caractéristique, en allant du bleu foncé pour les valeurs faibles à rouge foncé pour les valeurs élevées. Cela permet de visualiser rapidement quelles caractéristiques sont les plus importantes et les plus influentes pour le modèle, en se basant sur la valeur SHAP de chaque caractéristique.

### Limites et améliorations possibles

La modélisation réalisée dans le cadre de ce projet est basée sur une hypothèse forte concernant la définition d'une métrique d'évaluation. Cependant, cette métrique est construite à partir de certaines hypothèses qui ne peuvent pas être confirmées par les équipes métier de l'entreprise. Cette situation peut potentiellement limiter la pertinence de la métrique, et donc de la modélisation.

Pour remédier à cela, il serait important de travailler en étroite collaboration avec les équipes métier pour définir plus finement la métrique d'évaluation. En effet, une telle collaboration permettrait d'obtenir une compréhension plus claire des besoins des utilisateurs et des attentes de l'entreprise, ce qui pourrait contribuer à affiner la définition de la métrique d'évaluation utilisée. De cette manière, la modélisation pourrait être ajustée et améliorée pour répondre de manière plus précise aux exigences de l'entreprise et des utilisateurs.

Pour améliorer l'interprétabilité du modèle, il est suggéré de regrouper les variables obtenues à partir de la technique de "one hot encoding" en une seule variable lors de la perturbation. En effet, dans le cadre du jeu de données initial, un client ne peut pas cumuler plusieurs caractéristiques, ce qui implique que ces variables soient liées les unes aux autres. Cette modification permettrait d'améliorer la compréhension du modèle en simplifiant l'analyse des caractéristiques qui influencent les prédictions

Le traitement préalable du jeu de données a été réalisé de manière superficielle en utilisant un notebook qui ne prend en compte qu'une table du jeu de données. Cela offre probablement une occasion d'améliorer le modèle en utilisant d'autres fonctionnalités des données fournies, ainsi qu'en créant de nouvelles fonctionnalités en collaboration avec les équipes métier. Par conséquent, une analyse plus approfondie des données pourrait être effectuée pour identifier les caractéristiques pertinentes et potentiellement importantes, qui peuvent aider à améliorer la performance et l'interprétabilité du modèle