

---

# NOTE METHODOLOGIQUE

---

Implémentez un modèle de scoring

LECERF DEFER AMANDINE  
PARCOURS DATA SCIENTIST

## Table des matières

<b>I) Le contexte.....</b>	<b>1</b>
<b>II) Les données.....</b>	<b>1</b>
<b>III) Traitement des données.....</b>	<b>1</b>
<b>IV) Méthodologie : entraînement de modèles.....</b>	<b>2</b>
a) Choix d'algorithmes .....	2
b) Division des données .....	2
c) Choix des hyperparamètres de chaque algorithme.....	2
d) Contrecarrer le déséquilibre .....	2
<b>V) Choix d'un algorithme adapté.....</b>	<b>3</b>
a) Fonction coût : métrique métier .....	3
b) Métriques d'évaluation .....	3
c) Coefficient optimal .....	4
d) Seuil de probabilité .....	4
<b>VI) Interprétabilité.....</b>	<b>5</b>
a) Importance des features .....	5
b) Conséquence sur l'accord de prêt pour un client .....	6
<b>VII) Limites et améliorations possibles.....</b>	<b>7</b>

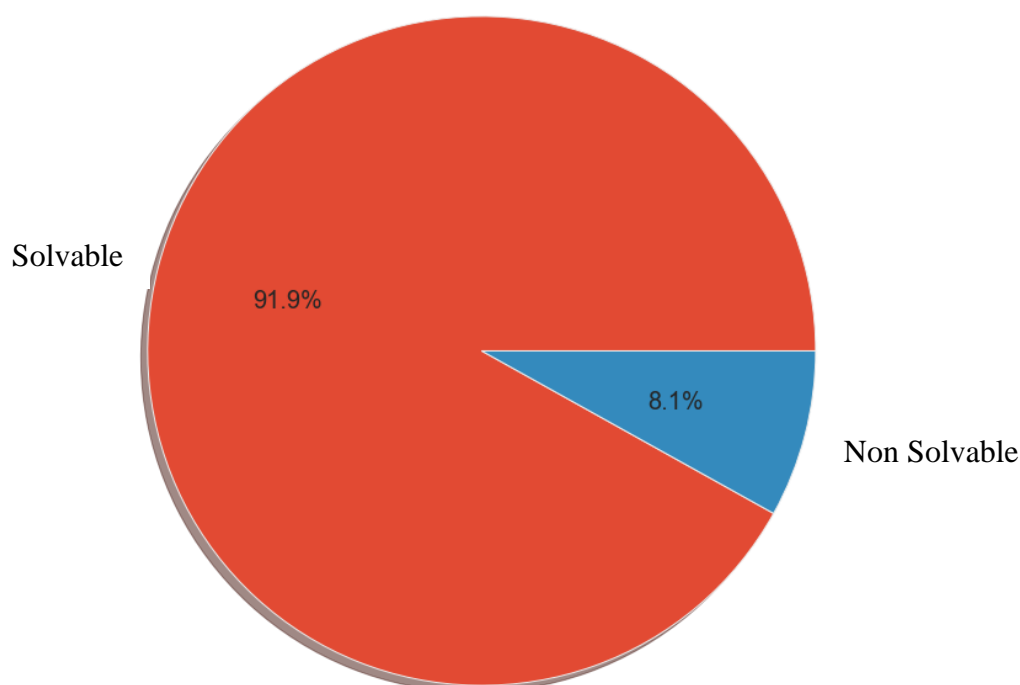
## I) Le contexte

La société financière nommée "Prêt à dépenser" propose des crédits à la consommation pour des personnes ayant peu ou pas d'historique de prêt. Cette entreprise souhaite développer un outil utilisant un modèle de scoring permettant d'obtenir la probabilité de défaut de paiement pour appuyer la décision d'accorder ou non un prêt à un client potentiel en s'appuyant sur des sources de données variées (données comportementales, données provenant d'autres institutions financières, etc.). Cet outil sera accompagné d'un dashboard interactif permettant d'interpréter les prédictions faites par l'outil. Prédire si un client remboursera ou non un prêt ou s'il aura des difficultés à le faire est un besoin commercial essentiel.

## II) Les données

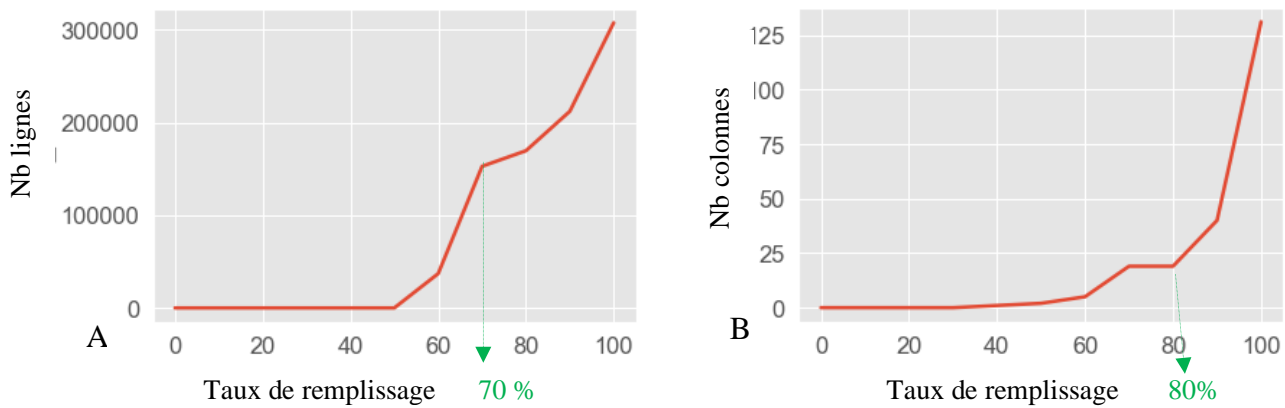
Pour réaliser cet outil, c'est le service dédié à la fourniture de crédits à la population non bancarisée, Home Credit, qui fournit une base de données. Cette base de données regroupe des informations générales pour chaque client ainsi que des informations supplémentaires sur chaque prêt contracté par les clients. N'étant pas du métier de la banque, je me concentre essentiellement sur les données générales de chaque client comme l'âge, le sexe, les revenus, l'emploi, le logement, les informations de crédit en cours, des notations externes, etc.

En observant de plus près la base de données, il apparaît un fort déséquilibre entre les personnes solvables et les personnes non solvables. Cette observation est représentée sur la figure ci-dessous.



## III) Traitement des données

Avant d'utiliser et d'exploiter au mieux les données à des fins de prédictions, le prétraitement de celles-ci est une tâche essentielle qui intervient après l'exploration des données. Il s'agit de plusieurs étapes de nettoyage des données brutes qui sont souvent peu fiables et incomplètes et leur utilisation peut générer des résultats faussés. Dans un premier temps, il est possible de les enrichir par de nouvelles variables (compréhensibles par des personnes qui ne sont pas du domaine bancaire) à partir des données déjà existantes comme le pourcentage du montant des crédits sur la totalité des revenus, la durée totale du paiement des crédits, ... De nombreux clients ont un important nombre de données manquantes. De ce fait pour ces clients nous disposons de peu de données et il est préférable de supprimer ces clients peu fiables (moins de 70% de remplissage). Cette même démarche peut être faite pour les colonnes (moins de 80% de remplissage). Cependant il faut veiller à garder suffisamment d'informations.



Après cette réduction de données, il est nécessaire d'imputer les données manquantes restantes, effectuer une standardisation des données pour finir par un encodage numérique des variables catégorielles.

## IV) Méthodologie : entraînement de modèles

### a) Choix d'algorithmes

Un modèle de prédiction prend en entrée des données et donne en sortie une conclusion.

Pour déterminer l'algorithme optimal de classification adapté à la problématique, quatre algorithmes ont été testés.

Un algorithme de régression binomiale (régression logistique comme baseline), un algorithme de forêt d'arbre aléatoire (random forest) et deux algorithmes de gradient boosting (XGBoost et Light-GBM).

### b) Division des données

Dans le cas d'une prédiction supervisée, l'entraînement d'un modèle nécessite une étape de division du jeu de données entre le prétraitement des données et la modélisation. Cette division permet de créer un échantillon d'apprentissage (train) qui représente 80% des données et un échantillon test qui représente 20% des données.

### c) Choix des hyperparamètres de chaque algorithme

Pour chaque algorithme à tester, une recherche d'hyper paramètres doit être effectuée pour que le modèle soit le plus adapté aux données. L'échantillon d'apprentissage est l'échantillon principal qui permet à un modèle d'ajuster sa prédiction par le biais d'une validation croisée. Durant cette étape, cet échantillon est séparé en plusieurs sous-ensembles sur lesquels les algorithmes apprennent pour ajuster le modèle. L'échantillon test, quant à lui, joue un rôle important car il permet d'évaluer les performances d'un modèle en comparant les prédictions du modèle aux vraies valeurs de la variable à expliquer. La version du modèle avec le score d'erreur minimisé est celui à garder pour la suite avec les paramètres optimums.

### d) Contrecarrer le déséquilibre

Comme mentionné précédemment, dans les données il y a un fort déséquilibre entre les clients solvables et non solvables. Ce déséquilibre va avoir un impact sur la performance de l'algorithme et les prédictions seront faussées car le modèle attribuera beaucoup plus fréquemment la classe la plus représentée aux clients. Pour pallier à ce problème de déséquilibre, quatre approches peuvent être appliquées sur les données d'entraînement :

- Class weights est une méthode directement gérée par les modèles et qui permet de pénaliser les poids associés aux observations de la classe sur-représentées (ici solvable).
- Over-sampling est une méthode qui va dupliquer aléatoirement des données existantes de la classe sous-représentée pour que chaque classe ait le même nombre de données que la classe sur-représentée à l'origine.
- SMOTE est une méthode qui va créer de nouvelles données pour la classe sous-représentée à partir des données existantes (et donc de la variété) pour que chaque classe ait le même nombre de données que la classe sur-représentée à l'origine.
- Under-sampling est une méthode qui va sélectionner une partie des observations sur-représentées pour que chaque classe ait le même nombre de données que la classe sous-représentée à l'origine.

## V) Choix d'un algorithme adapté

### a) Fonction coût : métrique métier

Pour déterminer l'algorithme optimal, chacun des quatre modèles a été entraîné selon les différentes méthodes pour pallier au déséquilibre. Les classes prédites sont comparées aux classes réelles et il est possible de créer une matrice de confusion pour se représenter les erreurs commises par le modèle :

		Classe réelle	
		Negatif (0)	Positif (1)
Classe prédite	Negatif (0)	TN	FP
	Positif (1)	FN	TP

Dans cette représentation, la classe 0 correspond aux clients solvables qui sont négatifs au refus de l'accord de prêt contrairement aux clients non solvables de la classe 1 qui sont positifs au refus de l'accord de prêt.

Ainsi :

- \* Accorder un crédit à un client ne pouvant pas le rembourser par la suite (FN) est synonyme de perte.
- \* Accorder un crédit à un client qui le remboursera par la suite (TN) est un gain.
- \* Ne pas accorder le prêt et que le client ne peut pas rembourser (TP) n'est ni une perte, ni un gain.
- \* Ne pas accorder le prêt alors que le client pouvait rembourser (FP) est une perte de client donc d'argent.

Une société de crédit cherche à gagner de l'argent et à "maximiser" ses gains. Il est donc possible de créer pour chaque modèle un score gain total en pondérant chaque cas par un coefficient qui pénalisera négativement les prédictions néfastes à la banque (FN et FP) et un qui valorisera les prédictions bénéfiques pour la banque (TN). Ce score sera alors normalisé par une normalisation min-max feature scalling :

$$gain\_total = (TN * coeff\_tn + FP * coeff\_fp + FN * coeff\_fn + TP * coeff\_tp)$$

$$gain = \frac{(gain\_total - gain\_min)}{(gain\_max - gain\_min)}$$

Avec :

$$gain\_min = (TN + FP) * coeff\_fp + (TP + FN) * coeff\_fn$$
$$gain\_max = (TN + FP) * coeff\_tn + (TP + FN) * coeff\_tp$$

Coefficients arbitraires possibles en respectant la métrique métier :

- \* FN ==> perte d'argent pour la banque ==> -100
- \* TP ==> refus de prêt ==> 0
- \* TN ==> prêt accordé, gain d'argent pour la banque ==> +10
- \* FP ==> client perdu, perte d'argent pour la banque ==> -1

Cette fonction coût a été implémentée afin de pénaliser l'impact des erreurs sur la décision d'octroi de crédit.

### b) Métriques d'évaluation

Pour analyser les résultats produits et ainsi démontrer l'efficacité de chaque modèle, il est nécessaire de se baser sur différentes métriques :

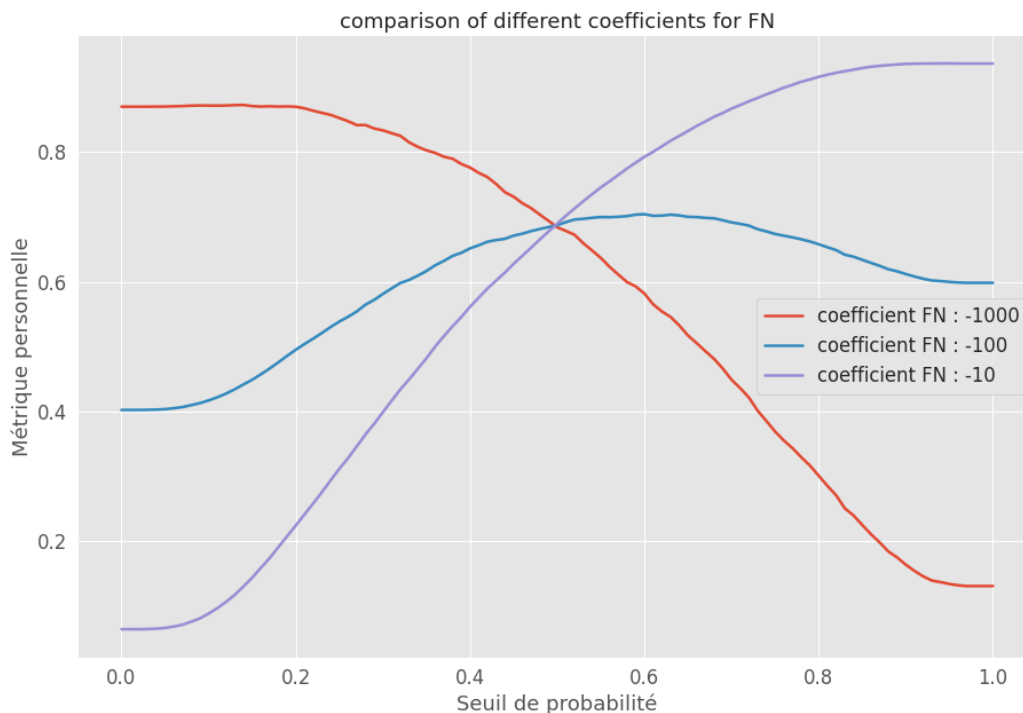
- L'accuracy est le ratio entre le nombre total de prédictions correctes et le nombre total de prédictions.
- La Precision est le ratio entre les vrais positifs et tous les positifs, il donne des indications sur les faux positifs.
- Le Recall est la proportion dans laquelle notre modèle identifie correctement les vrais positifs.
- Le score F1 est la moyenne harmonique de la précision et du rappel. Un bon score F1 indique une bonne précision et une bonne valeur de rappel.
- L'AUC (l'aire sous la courbe ROC) est la mesure de la capacité d'un classificateur à distinguer les classes et correspond au résumé de la courbe ROC. Plus l'AUC est élevée, plus la performance du modèle à distinguer les classes positives et négatives est bonne.

Le modèle utilisé pour l'outil doit maximiser l'AUC, le score F1, le Recall. Une banque cherche à ne pas accorder de prêt à des clients qui ne peuvent pas le rembourser (TP à maximiser). La banque ne veut pas accorder un prêt à un client qui ne peut pas le rembourser (FN à minimiser).

Chaque modèle doit donc maximiser ce gain et remplir les contraintes des métriques d'évaluation. Toutes ces conditions sont validées par le modèle Light Gradient Boosting Machine (modèle robuste qui repose sur le principe d'arbres de décisions) avec la méthode de l'under-sampling pour pallier au déséquilibre qui est donc ici le modèle optimal.

### c) Coefficient optimal

Le seuil de probabilité est la valeur de prédiction à partir de laquelle le modèle va considérer qu'un client est solvable ou non compte tenu de la fonction gain employée. Des coefficients arbitraires ont précédemment été définis mais ceux-ci ne sont peut-être pas optimaux. Le coefficient de FN est celui qui a la plus grande importance dans notre métrique mais le coefficient a-t-il le bon ordre de grandeur. Pour vérifier cela, le score gain a été recalculé avec le modèle optimal pour différent seuil de probabilité en variant l'ordre de grandeur du coefficient FN :



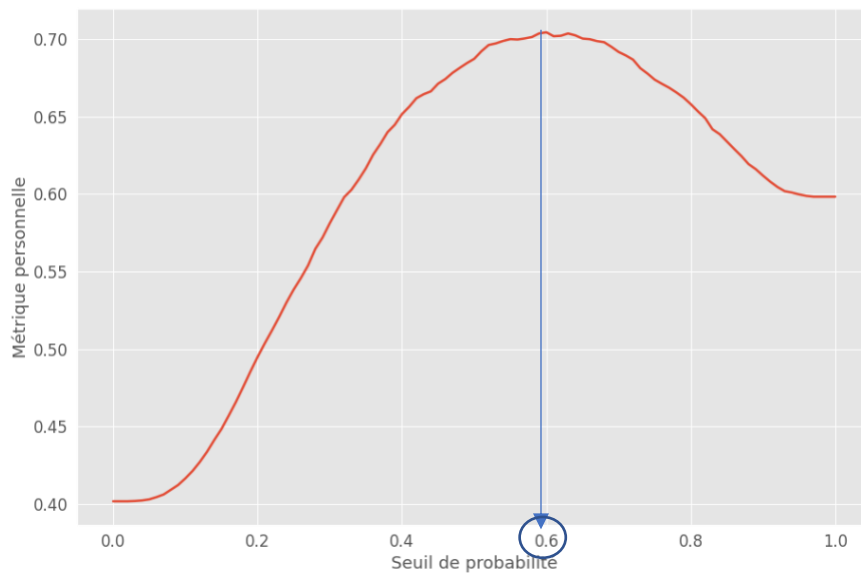
Avec un coefficient de -1000, le seuil optimal pour que la banque ait le maximum de gain d'argent se situe aux alentours d'une probabilité de 0,2 à partir de laquelle on peut voir que la courbe des gains décroît rapidement. Ce coefficient est trop strict car seules les personnes avec une probabilité de défaut de paiement  $< 0,2\%$  se voient accepter leur prêt ce qui correspond à peu de personnes.

Au contraire, avec un coefficient de -10, la banque est trop laxiste dans le choix des clients pour l'accord d'un prêt car le seuil optimal pour que la banque ait le maximum de gain d'argent se situe aux alentours d'une probabilité de 0,9 et toutes les personnes avec une probabilité de défaut de paiement  $< 0,9\%$  se verraient accepter leur prêt ce qui correspond à beaucoup de personnes.

Un coefficient de -100 est mieux adapté car le seuil optimal se situerait aux alentours de 0,55 ce qui est correct pour déterminer si oui ou non le prêt est accepté.

### d) Seuil de probabilité

La métrique personnelle gain doit être maximisée pour que la banque gagne le plus d'argent et ainsi permettre de déterminer à partir de quel seuil de probabilité un client est considéré comme solvable ou non. Généralement le seuil de probabilité choisi par les modèles pour classer les individus est de 0,5 mais au vu de la courbe du score gain (avec un coefficient de -10), il ne s'agit pas du seuil optimal pour que la banque maximise ses gains.

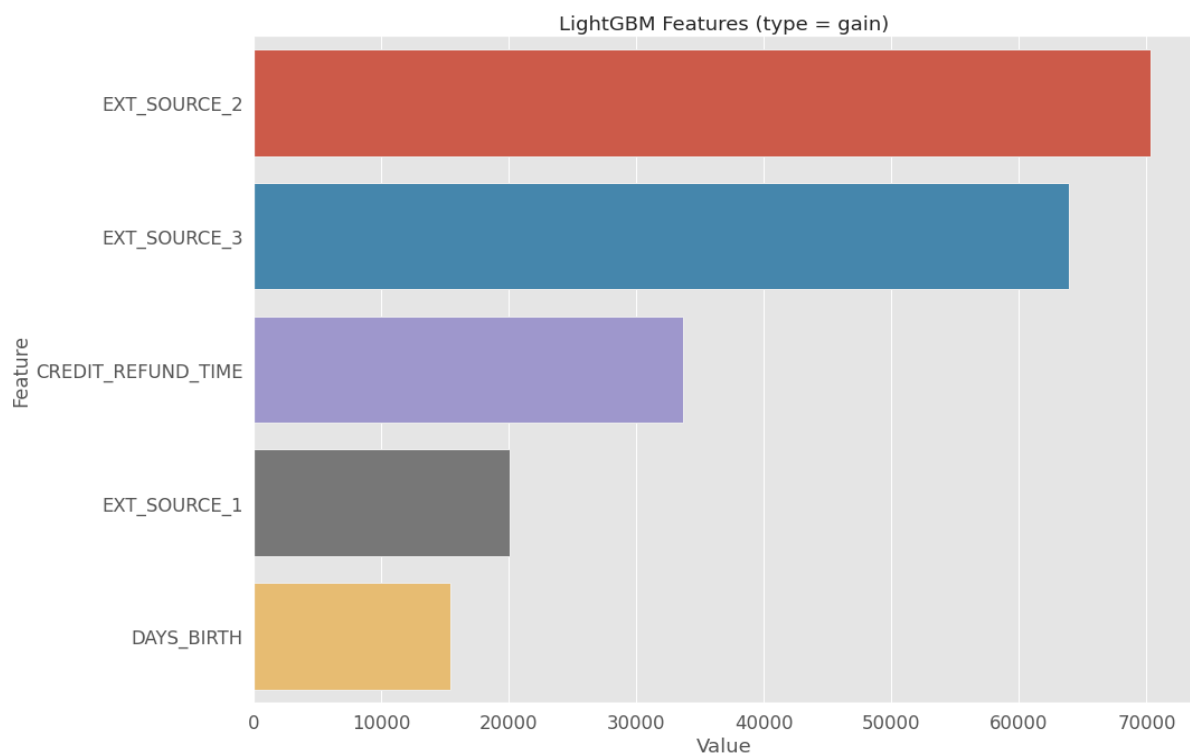


Les résultats montrent un seuil de 0,6 ce qui signifie qu'un client avec une prédiction au-dessus de cette valeur sera catégorisé comme non solvable et un client avec une prédiction en dessous de cette valeur sera considéré comme solvable.

## VI) Interprétabilité

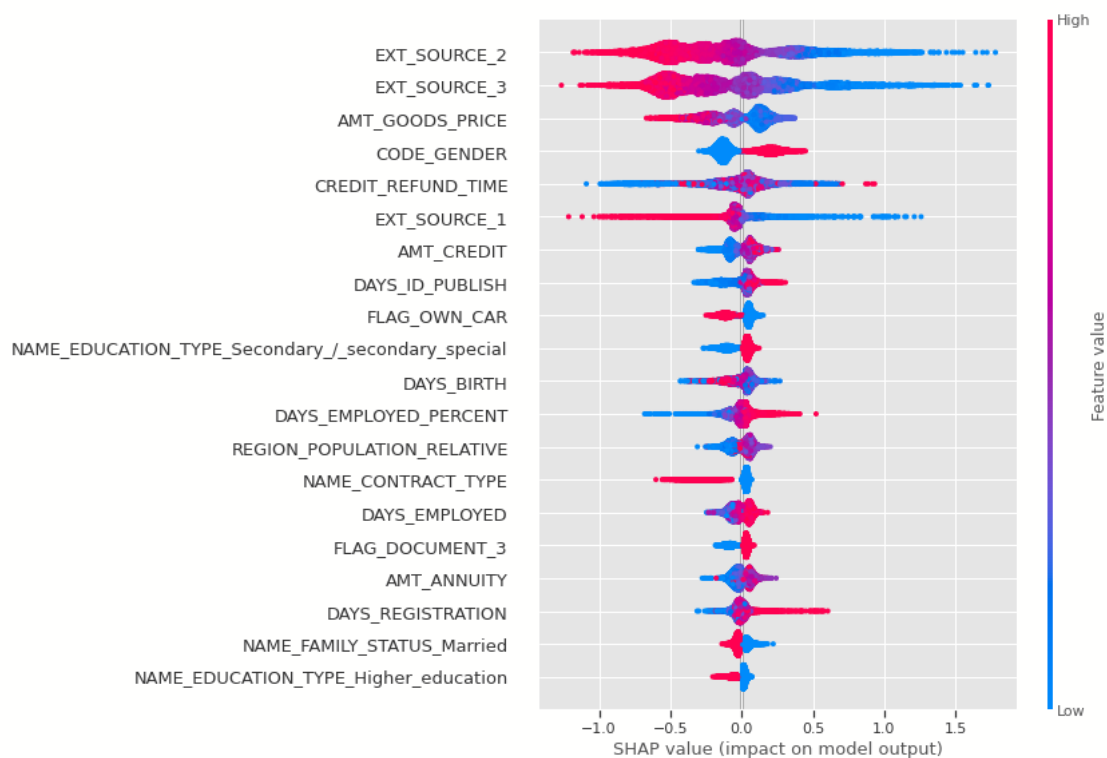
### a) Importance des features

Maintenant que le modèle, les coefficients et le seuil de solvabilité ont été définis, il est désormais intéressant de savoir quelles sont les informations qui ont un poids important dans le calcul de la probabilité de solvabilité d'un client. Le modèle LGBM est un modèle basé sur des arbres et l'importance des features est donnée par la fonction `feature_importance` avec pour option `importance_type='gain'`. L'importance des features se base sur la réduction moyenne de la perte obtenue lors de l'entraînement du modèle.



Avec cette représentation, on peut dire que les features les plus importantes pour la prédiction d'accord d'un prêt sont les sources extérieures 2 et 3 qui sont des scores normalisés créés à partir de sources de données externes. Puis on trouve la durée que va mettre un client à rembourser un prêt en année ainsi que la source externe 1. Enfin le nombre de jours depuis la naissance des clients donc leur âge joue un rôle important dans l'acceptation d'un crédit.

Pour plus de précision et pour connaître le sens d'influence de chacune des variables, il est possible d'utiliser la librairie SHAP (SHapley Additive exPlanations) qui explique la sortie de tout modèle d'apprentissage automatique en utilisant la théorie des jeux. Lorsque l'on regarde globalement l'importance des features avec l'outil SHAP, nous retrouvons les cinq features déterminées précédemment par la fonction intégrée dans notre modèle ainsi que les features représentant le sexe du client et le prix du bien acheté par des prêts de consommation qui sont, selon moi, des variables discriminantes et que j'ai décidé de ne pas prendre en compte. SHAP nous apporte des informations sur le sens de l'importance de chaque feature.



Nous traçons les valeurs SHAP de chaque caractéristique pour chaque échantillon sur l'axe des x, puis nous les laissons s'accumuler. Si nous colorons ensuite chaque point selon sa valeur, nous pouvons voir comment une valeur faible ou élevée affecte la sortie du modèle.

Pour EXT\_SOURCE\_2, EXT\_SOURCE\_3, EXT\_SOURCE\_1 on peut voir que de faibles valeurs augmentent de manière significative la sortie de probabilités du modèle et donc le fait d'être non solvable et le non accord du prêt.

Au contraire, pour CREDIT\_REFUND\_TIME, plus le temps de remboursement des crédits est grand plus il y a de chance que le prêt ne soit pas accordé.

### b) Conséquence sur l'accord de prêt pour un client

Le sens de l'importance de chaque feature est connu par l'outil SHAP mais ne connaissant pas le domaine bancaire, il ne m'est pas possible de vérifier si les graphiques précédents sont cohérents avec ce domaine.

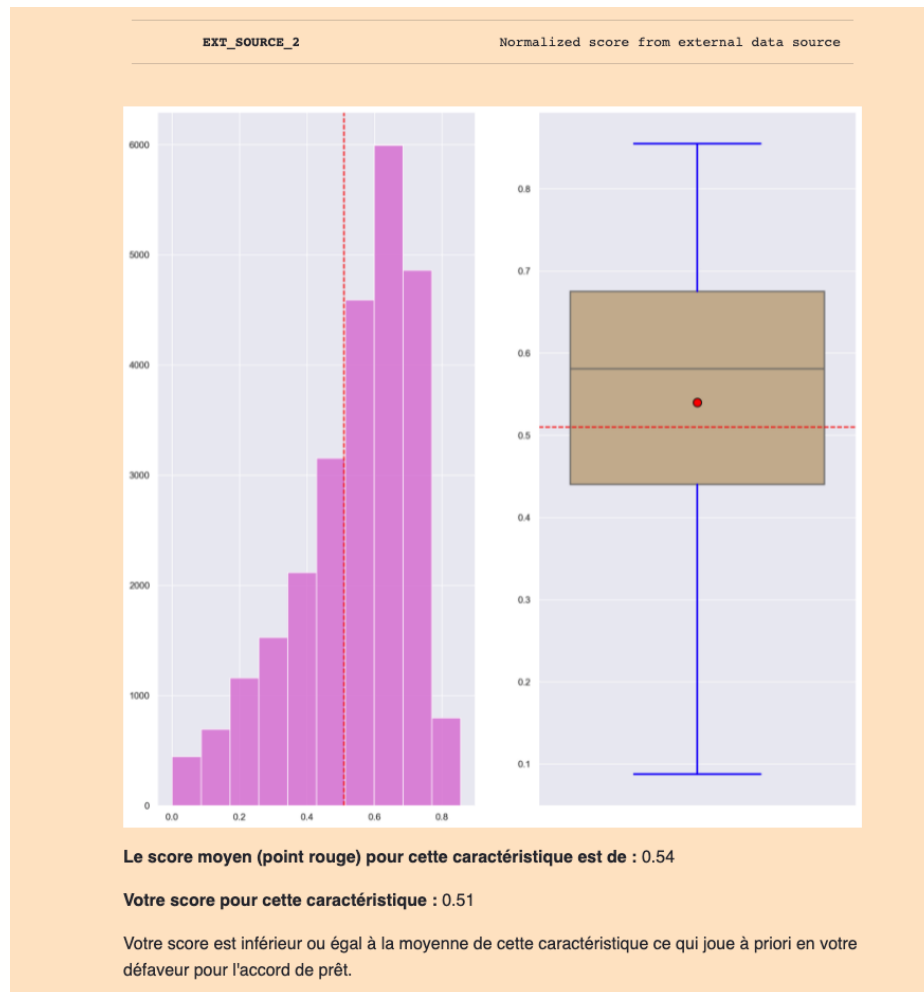
Ces hypothèses ont été implémentées dans un dashboard (<https://dashboard-p7.herokuapp.com/>). Pour celui-ci, je décide de ne prendre que des identifiants qui n'ont pas servis à l'établissement du modèle et qui ne possèdent pas l'information de solvabilité car il s'agit de l'élément recherché par l'outil demandé par la banque.

Pour les variables EXT\_SOURCE 1, 2, 3, plus ces scores seraient faibles et plus le client risquerait de ne pas voir sa demande de prêt acceptée.

Au vu des observations précédentes, pour chacune de ces variables, il serait possible de dire que si un client à un score inférieur à la moyenne de cette variable alors celle-ci jouerait en sa défaveur pour l'accord du prêt bancaire.

Pour la durée que met un client à rembourser son prêt, on peut supposer que plus le temps de remboursement des crédits est grand plus le client a de chance que le prêt ne lui soit pas accordé.

Pour cette variable, il serait possible de dire que si un client à un score supérieur à la moyenne de cette variable alors celle-ci variable jouerait en sa défaveur pour l'accord du prêt bancaire.



Enfin, concernant l'âge des clients, comme nous l'avons vu lors du preprocessing, les jeunes clients ont plus de mal à rembourser leur prêt. Pour cette variable, il serait possible de dire que si l'âge d'un client est inférieur et éloigné du 1<sup>er</sup> quartile (25% des clients) de l'âge alors cette variable jouerait en sa défaveur pour l'accord du prêt bancaire.

## VII) Limites et améliorations possibles

La modélisation a été effectuée sur la base d'une métrique personnelle créée pour répondre au mieux au besoin de gain d'argent d'une banque. Les coefficients de cette métrique ont été choisis arbitrairement selon le bon sens. L'axe principal d'amélioration serait donc de définir plus précisément ces coefficients associés à chaque combinaison classe prédite/classe réelle car le modèle déterminé ici ne sera pas obligatoirement le meilleur.

Le modèle créé ici tend à être un modèle "éthique" car il ne tient pas compte de variables telles que le genre du client qui peut être une information discriminante dans l'attribution d'un prêt. Cependant toutes les variables discriminantes n'ont peut-être pas été enlevées. Il serait intéressant de comparer un modèle "éthique" à un modèle prenant en compte toutes les variables pour voir si le fait de supprimer ces variables discriminantes amène une perte d'information et de précision pour les prédictions ainsi qu'une perte de rentabilité pour la banque.

Le traitement des données a été réalisé de façon superficielle car une seule table de données a été prise en compte alors que des données sur les précédents prêts étaient disponibles. Il y a donc très probablement la possibilité d'améliorer la modélisation en utilisant d'autres features provenant des autres bases de données ainsi qu'en créant de nouvelles features grâce à l'équipe métier.

Enfin, le Dashboard interactif pourra être amélioré pour répondre au mieux aux attentes et besoins des conseillers clients.