Projet

Implémentation d'un modèle d'apprentissage supervisé pour une application de **Crédit Scoring**.

Objectifs

- Le modèle doit permettre de définir la probabilité de défaut de remboursement d'un crédit sur la base d'informations relatives au client.
- Il doit également offrir un certain niveau de transparence concernant les données et leurs traitements en vue d'implémenter des méthodes d'interprétabilité des variables.

Données

- Le jeu de données est constitué d'informations relatives aux crédits en cours et autres informations externes. Une partie du jeu concerne des crédit échus et comprend donc également des étiquettes, c'est-à-dire une valeur binaire indiquant si le crédit a été remboursé ou non. Ce jeu est utilisé pour l'apprentissage automatique.
- Une autre partie du jeu ne contient pas d'étiquette, il s'agit des dossiers en cours pour lesquels il faut réaliser un classement prédictif avec le modèle précédemment entraîné.

Modélisation

Le résultat attendu le plus important pour un dossier est la valeur de la **probabilité de non-paiement**. En appliquant un **seuil** à cette valeur, nous pouvons lui affecter une **valeur binaire** (0 ou 1) suivant que la probabilité est inférieure ou supérieure au seuil.

Si la probabilité est inférieure au seuil, on considère que le crédit sera remboursé, le dossier est négatif (0). Inversement, si le probabilité est supérieure au seuil, on considère que le crédit ne sera pas remboursé, le dossier est positif (1).

➢ Il s'agit donc d'un problème de classification dont le résultat binaire dépend du paramètre: **seuil de classification**.

Pour évaluer le modèle on peut comparer les valeurs prédites avec les valeurs réelles. On peut ainsi établir un tableau indiquant le nombre de valeurs correctement prédites ou non (matrice de confusion).

				Classes réelles			
				remboursé		non remboursé	
				0	negatif	1	positif
Classes prédites	remboursé	0	négatif	TN	**42116**	FN	**1586**
	non remboursé	1	positif	FP	**14405**	TP	**3396**

42116 dossiers sont prédits négatifs et le sont réellement (True Negative - **TN**)
1586 dossiers sont prédits négatifs mais sont positifs en réalité (False Negative - **FN**)
14405 dossiers sont prédits positifs mais sont négatifs en réalité (False Positive - **FP**)
3396 dossiers sont prédits positifs et le sont réellement (True Positive - **TP**)

Le tableau est basé sur des valeurs binaires (0 ou 1) déterminées selon un **seuil de classification** donné. On obtiendrait d'autres valeurs en appliquant un seuil différent.

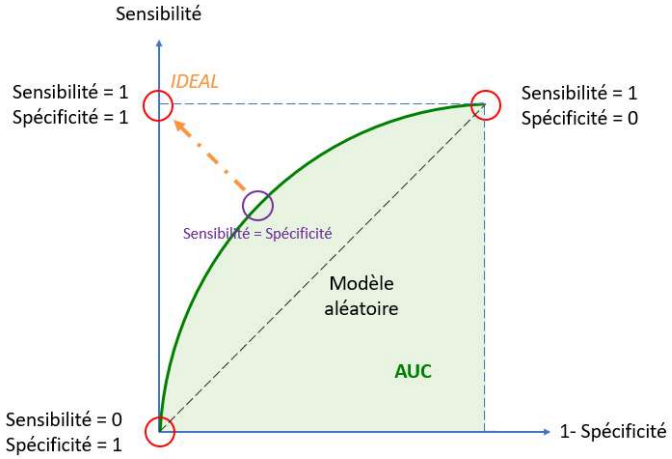
Ces valeurs peuvent être traduites en des indicateurs caractérisant le modèle, dont voici les plus importants:

Sensibilité = TP / (TP + FN) = 3396 / (3396 + 1586) = 0,6817
Capacité du modèle à détecter tous les dossiers de crédit non remboursés (1)

Spécificité = TN / (TN + FP) = 42116 / (42116 + 14405) = 0,7451
Capacité du modèle à détecter les dossiers de crédit remboursé (0)

Précision = TP / (TP + FP) = 3396 / (3396 + 14405) = 0,1908
Capacité du modèle à détecter les vrais dossiers non remboursés (1)

Nous pouvons représenter les évolutions de la **sensibilité** et de la **spécificité** en fonction du **seuil de classification** avec une **courbe ROC**.



Une **courbe ROC** caractérise le classifieur qui a produit les résultats sous forme de probabilités par variation du **seuil de classification**. Un modèle idéal a une **sensibilité** et une **spécificité** = **1**. Plus la courbe se rapproche de cet idéal, meilleurs sont les indicateurs.

- On peut résumer la mesure de cette performance par l'aire sous la courbe (**A**rea **U**nder the **C**urve).

Classifieurs

La mesure **AUC** est donc utilisée pour l'évaluation des modèles et représente à ce titre un critère de sélection du classifieur.
Le classifieur doit bien entendu intégrer des méthodes de classification binaire. Il doit être capable de traiter un nombre important de données dans un temps d'exécution raisonnable. Enfin, il doit offrir des fonctionnalités permettant l'interprétabilité des variables.

Les types de classifieurs adaptés sont les algorithmes basés sur les arbres de décision. Nous avons testé les modèles suivants: Random Forest, XGBoost, CatBoost et LightGBM. Le jeu d'entraînement de départ contient **246008** dossiers et **897** variables.

- La classifieur répondant le mieux à l'ensemble des critères est LightGBM avec une mesure **AUC** parmi les meilleures (**0.782**).

Optimisation technique

Afin d'augmenter les performances, on a réduit le nombre de variables à **450** avec la méthode **RFECV**.

- Le modèle retenu entraîné sur le jeu réduit affiche des résultats équivalents (**AUC** = **0.781**).
- Nous avons ensuite optimisé les paramètres du classifieur avec la méthode **Hyperopt** et obtenu un **AUC** de **0.787**.

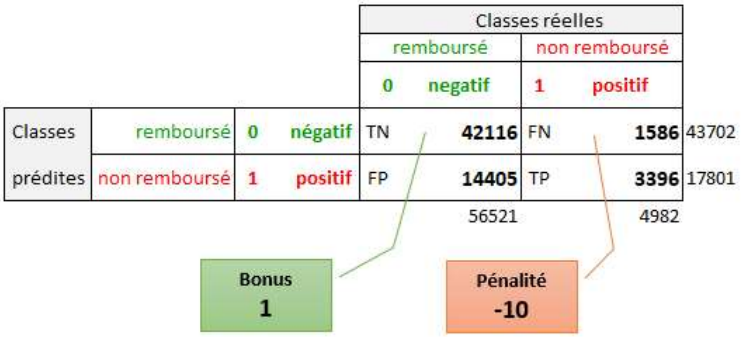
Optimisation métier

Optimiser la valeur de la mesure **AUC** permet d'améliorer globalement la sensibilité et la spécificité. Cette approche est pertinente si on considère les éléments de la matrice de confusion de même importance.

Dans le domaine bancaire, un crédit non remboursé coûte plus cher qu'un dossier de crédit non signé. Il s'agit de trouver le meilleur compromis entre le nombre de crédit qu'on accorde mais qui ne seront *in fine* pas remboursés (les faux négatifs) et le nombre de crédit qu'on refuse et dont on perd potentiellement le bénéfice sur les intérêts pour les clients solvables (les faux positifs).

- On peut alors définir une **fonction de coût** en accordant des poids différents aux éléments de la matrice de confusion.

Nous avons ainsi affecté un poids de -10 à chaque dossier prédit négatif mais réellement positif (FN) et un poids de +1 aux dossiers négatifs identifiés comme tels (TN). Les poids des dossiers FP et TP sont nuls.



Le gain résultant pour une valeur donnée du paramètre **seuil de classification** est alors défini par la **fonction d'évaluation du gain** suivante:

$$Gain = TP * (0) + TN * (1) + FP * (0) + FN * (-10)$$

L'optimisation métier du classifieur consiste à maximiser le **Gain** (normalisé pour la circonstance) en faisant varier le paramètre **seuil de classification** mais également les paramètres du classifieur.

Cette opération est réalisée avec la méthode **Hyperopt** en passant à la fonction ***objective*** un dictionnaire de paramètres avec des gammes de valeurs et pour métrique d'évaluation le **Gain normalisé** à maximiser.

- L'optimisation métier résulte en un **Gain normalisé** de **0.711** et une valeur **AUC** de **0.786**, donc un peu moins bonne que la valeur obtenue avec l'optimisation technique.

Interprétation du modèle

Importance des variables

Les modèles testés offrent tous une méthode permettant d'extraire les variables ayant le plus d'influence sur la classification. Un score leur est affecté à cet effet.

Nous avons utilisé cette fonctionnalité pour la ***sélection de variables*** dans le cadre de l'optimisation du modèle par ***réduction de dimensions***. Plus précisément nous avons exploité les résultats obtenus avec la méthode **RFECV** pour réduire les variables de moitié.

- Outre les gains en temps de calculs obtenus, cette approche permet également de disposer d'une information relative à l'importance globale des variables dans le modèle, qui peut être débattue et validée avec les experts *métier*.

	Variable	Importance
0	CNT_CHILDREN	1
1	AMT_INCOME_TOTAL	1
2	AMT_CREDIT	1
3	AMT_ANNUITY	1
4	REGION_POPULATION_RELATIVE	1

Variables sélectionnée avec RFECV

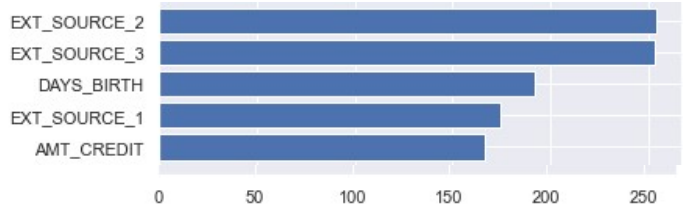
	Variable	Importance
107	OCCUPATION_TYPE_Medicinestaff	2
108	OCCUPATION_TYPE_Privateservicestaff	2
109	OCCUPATION_TYPE_Realtyagents	2
111	OCCUPATION_TYPE_Secretaries	2
112	OCCUPATION_TYPE_Securitystaff	2

Variables exclues (peu d'influence)

Interprétabilité globale

Le jeu de données à **450 variables** a été utilisé pour l'optimisation des paramètres du classifieur **LightGBM** avec pour objectif la maximation du gain selon la **fonction d'évaluation**.

Ci-dessous, un extrait des variables importantes classées selon leur score, relatives à l'apprentissage du classifieur optimisé.



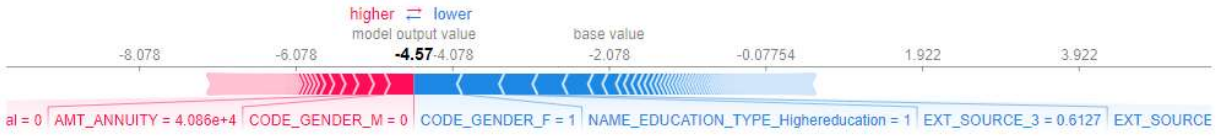
- Ces informations représentent un premier niveau d'interprétation du modèle. Elles indiquent quelles sont, globalement, les variables qui influencent le plus la décision de classification.

Interprétabilité locale

Nous avons implémenté la méthode **SHAP** (**SH**apley **A**dditive ex**P**lanations) qui consiste à calculer la *valeur de Shapley* pour toutes les variables de tous les individus c'est-à-dire la moyenne de l'impact d'une variable (sur la sortie, donc la prédiction) pour toutes les combinaisons de variables possibles.

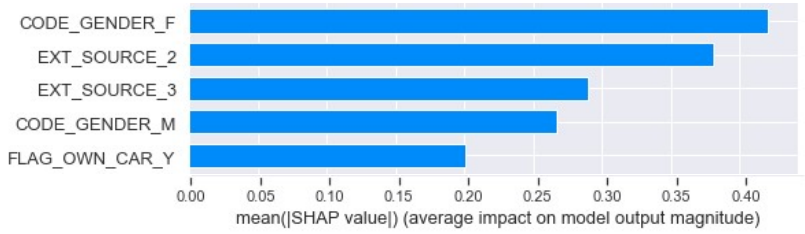
- La somme des effets de chaque variable explique alors la prédiction.

Le graphe ci-dessous représente les impacts des valeurs des variables importantes sur la prédiction. Les variables en rouge contribuent le plus à une prédiction positive (dossier non remboursé), les variables en bleu contribuent le plus à une prédiction négative (dossier remboursé).

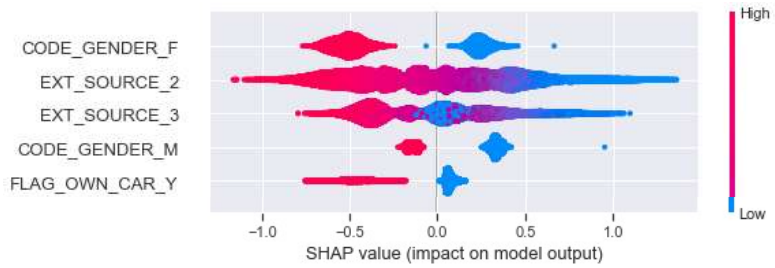


Dans cet exemple, la valeur SHAP est inférieure à la valeur de base, le dossier est classé négatif. Les variables en bleu expliquent cette décision.

SHAP permet également une ***analyse globale***. En effet, en moyennant les valeurs absolues des valeurs de chaque variable (niveau local), on remonte à l'importance globale des variables.



Ci-dessous : les dossiers sont représentés par des points dont la couleur varie en fonction de la valeur de la variable (petite à grande valeur = bleu à rouge). Les points sont positionnés sur l'axe des abscisses selon leur valeur SHAP ce qui caractérise l'impact de la variable sur la prédiction.



Le fait d'être une femme présente globalement un impact négatif sur la valeur SHAP ce qui contribue à ce que la prédiction soit plus basse que la valeur de base (ce qui caractérise les dossiers négatifs).

Amélioration du modèle

La réalisation du modèle a nécessité la conception de nombreux blocs de transformation et de traitement des données. Chaque bloc fait appel à des méthodes paramétrables. De fait, les résultats sont dépendants des paramètres choisis. L'architecture du code permet d'optimiser les blocs indépendamment.

Sélection des variables

Les informations disponibles relatives à l'importance des variables sont débattues avec les experts métier en vue de définir les stratégies techniques à tester dans les différents blocs concernés :

- Valeurs manquantes
- Corrélations entre variables
- Seuil de variance
- Réduction de dimensions (RFE)

Equilibrage des données

L'équilibrage des données introduit des données artificielles donc la possibilité d'incohérences. Des tests peuvent être réalisés en variant certains paramètres (ratios classes).

Fonction d'évaluation du gain

Les règles métier et les critères financiers relatifs aux pertes et profits doivent être communiqués en vue d'établir une fonction d'évaluation du gain adaptée.

Optimisation des hyper-paramètres

Nous avons testé plusieurs classifieurs et retenu **LightGBM** pour sa rapidité d'exécution. D'autres classifieurs comme **XGBoost** peuvent potentiellement apporter de meilleures performances techniques. Il s'agit de les tester dans le pipeline complet si les contraintes de temps en production le permettent.

Hyperopt implique le réglage de nombreux hyper-paramètres à optimiser, en particulier les gammes de valeurs des paramètres du classifieur et le nombre d'évaluations.

Interprétation

La méthode **SHAP** offre des résultats intéressants, en particulier pour l'interprétabilité locale. Par contre, son implémentation dans une application web (comme Dash) est encore difficile, il s'agit d'investiguer pur trouver des solutions plus performantes que celle adoptée (transfert d'image).

Concernant l'interprétabilité globale, les algorithmes basés sur les forêts aléatoires (mais également SHAP) permettent d'identifier les variables influentes. D'autres approches comme les permutations de variables peuvent être testées pour valider les résultats.

L'application **Dash** offre des fonctionnalités permettant de comparer des dossiers, ce qui participe à la compréhension du modèle. On pourra néanmoins optimiser la fonction de similarité basée sur la méthode des plus proches voisins.