

Openclassrooms

Projet 7

Implémentez un
modèle de scoring





Sommaire

I, Preprocessing des datas

- a, DF train et test
- b, Bureau and balances
- c, Previous application
- d, POS_cash_balance
- e, installments payment
- f, credit_card_balance
- g, Jonction des datas

II, Recherche d'un modèle de prédiction¶

- a, Mise en place d'un jeu d'entraînement et de test équilibré avec SMOTE
- b, Recherche d'un algorithme pertinent

Regression logistic

Random forest

Light GBM

- c, Amélioration de l'algorithme

Recherche d'hyperparamètres avec Halving SearchCV

Recherche du seuil pertinent

- d, Analyse de l'importance des features locales

Avec LIME

Avec SHAP

III, L'outils final : le dashboard

Comme il était conseillé de le faire dans l'énoncé, ce prétraitement des données reprends très largement le kernel partagé par **Home crédit group** (l'organisateur de la compétition).

En voici les principales étapes.

I, PRETRAITEMENT DES DATAS

a, Données ‘Train’ et ‘Test’

Dimensions : 307 511 x 122 et 48 744 x 121 (différence due à la colonne **TARGET**)

Description : données statistiques des clients.

Principales opérations effectuées :

- fusion
- suppression du genre non spécifié dans la colonne ‘CODE_GENDER’
- encodage binaire des colonnes à deux choix
- remplacement de la donnée aberrante dans ‘DAYS_EMPLOYED’
=>Nan
- création de nouvelles colonnes :
 - ‘DAYS_EMPLOYED_PERC’ durée d’emploi par rapport à l’âge de la personne
 - ‘INCOME_CREDIT_PERC’ revenus totaux sur montant du crédit
 - ‘ANNUITY_INCOME_PERC’ annuités par rapport aux revenus
 - ‘PAYMENT_RATE’ annuité par rapport au crédit



I, PRETRAITEMENT DES DATAS

b, Données ‘Bureau’ et ‘Balance’

Dimensions : 1 716 428 x 17 et 27 299 925 x 3

Description : historique des crédits contractés par les clients auprès d'institutions tierces et leur balances mensuelles.

Principales opérations effectuées :

- encodage des colonnes
- agrégation et regroupement des jeux de données par ‘SK_ID_BUREAU’
- création de colonnes avec MIN MAX MEAN VAR
- création des colonnes closed pour les crédits fermés



I, PRETRAITEMENT DES DATAS

c, Données ‘applications antérieures’

Dimensions : 1 670 214 x 37

Description : historique des applications auprès de Crédit loans.

Principales opérations effectuées :

- encodage des colonnes
- remplacement de la valeur aberrante 365,243 dans les colonnes contenant des jours en durées
- ajout de la colonne ‘APP_CREDIT_PERC’ montant demandé sur montant prêté
- création de colonnes avec MIN MAX MEAN VAR
- remise en une dimension de la colonne ‘NAME_CONTRAT_STATUS’ pour bien distinguer les données des crédits approuvés et refusés



I, PRETRAITEMENT DES DATAS

d, Données 'POS_cash_balances'

Dimensions : 10 001 358 x 8

Description : relevés de compte mensuels des clients ayant souscrit des crédits auprès de la compagnie.

Principales opérations effectuées :

- encodage des colonnes
- création de colonnes avec MIN MAX MEAN VAR



I, PRETRAITEMENT DES DATAS

e, Données ‘échelonnage des paiements’

Dimensions : 13 605 401 x 8

Description : historique des paiements pour des crédits accordés antérieurement auprès de la compagnie.

Principales opérations effectuées :

- encodage des colonnes
- création des colonnes ‘PAYMENT_PERC’ montant payé par rapport à l’échéancier et ‘PAYMENT_DIFF’ montant restant du.
- création des colonnes ‘DPD’ et ‘DBD’ (day past due et day before due) pour établir des données sur les retards ou avance de remboursement
- création de colonnes avec MIN MAX MEAN VAR



I, PRETRAITEMENT DES DATAS

f, Données 'crédit_card_balance'

Dimensions : 3 840 312 x 23

Description : relevés de carte mensuels des cartes de crédit que l'applicant avait chez Home Crédit

Principales opérations effectuées :

- encodage des colonnes
- agrégation par 'SK_ID_CURR'
- création de colonnes avec MIN MAX MEAN VAR



I, PRETRAITEMENT DES DATAS

g, Jonction des datas

Jonction des données sur la clef ‘SK_ID_CURR’.

Dimension du jeu de donnée : 356251 x 399

Suppression des lignes concernant les données ‘Test’ (sans valeur TARGET).

Remplacement des valeurs nulles des colonnes non-encodables par la moyenne de la colonne.

Normalisation des données dans un intervalle [0,1].

Suppression des colonnes n’ayant qu’une valeur identique pour tous les individus.

Jeu de données final de 307507 lignes et 392 colonnes.



Ce travail de **nettoyage** et de **jonction** des données est relativement long mais **fondamental** dans la mesure où il crée et sélectionne les **critères** sur lesquels nous allons entraîner notre modèle.

Un choix de critères différents entraînera le modèle différemment et produira des **conclusions autres**.

Il n'y a **pas de manière optimale** de faire ce travail. Sur Kaggle les jeux de données finaux font entre quelques dizaines et plus de 1000 colonnes.

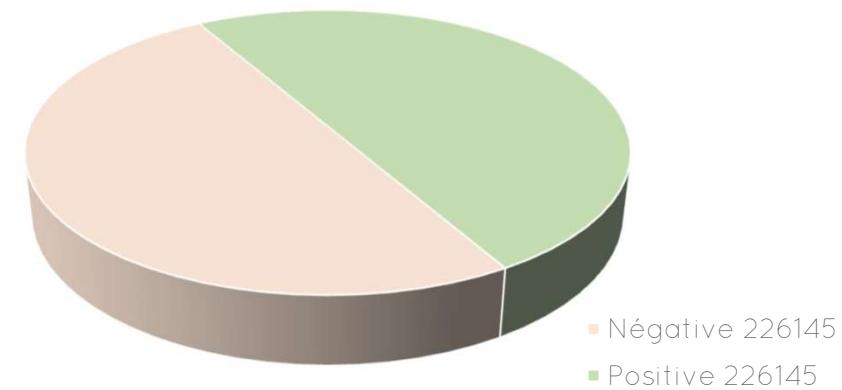
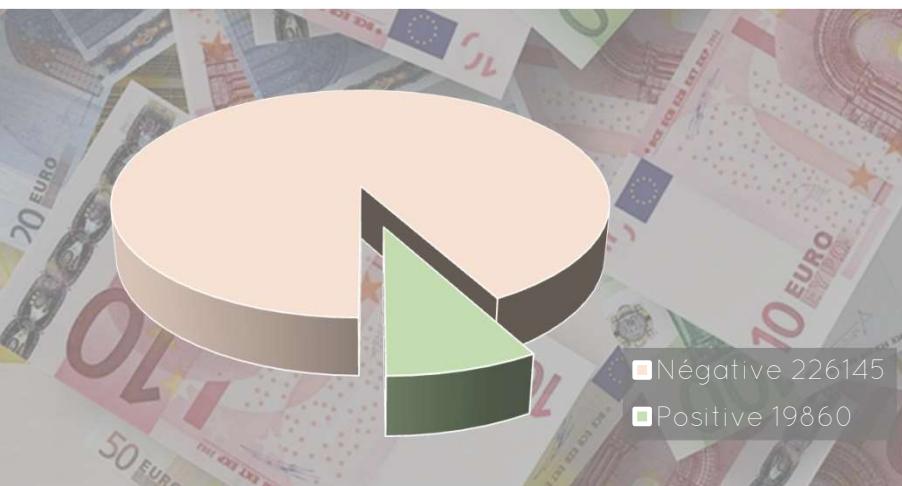
II, RECHERCHE D'UN MODÈLE DE PRÉDICTION

a, Mise en place d'un jeu d'entraînement équilibré avec SMOTE



Création d'un jeu de données de **test et d'entraînement**.

Fort **déséquilibre** entre les données positives et négatives :



SMOTE créé des individus de synthèse (pas des copies) à partir des individus de la classe minoritaire.

La méthode se base sur l'algorithme des plus proches voisins (KNN).

II, RECHERCHE D'UN MODÈLE DE PRÉDICTION

b, Recherche d'un algorithme pertinent



La mesure de l'efficacité algorithmique.

Ici nous recherchons un algorithme qui sera à même de performer sur une problématique de **classement binaire**.

La métrique de prédilection pour l'évaluation de ces algorithmes est la **courbe ROC** et l'aire sous la courbe (AUC).



En complément, et vu la spécificité de notre problématique, notamment concernant le poids des faux positifs, nous établissons également un **outil de scoring personnalisé** qui pénalise fortement ces **faux positifs**.

```
def scoring_perso(y_test, y_pred):
    cm = confusion_matrix(y_test, y_pred)
    tn, fp, fn, tp = confusion_matrix(y_test, y_pred).ravel()
    score2 = 1 - ((10*fp + fn - 5*tp) / cm.sum())
    return score2
```

Faux positif
erreur grave
coef 5

Faux négatif
erreur 'normale'

Vrai positif
encourage les
prédictions avec
bcp de TP.

II, RECHERCHE D'UN MODÈLE DE PRÉDICTION

b, Recherche d'un algorithme pertinent

Point **vocabulaire** pour la mesure de la **performance** des algorithmes de **classification**.

Sensibilité :

$$\frac{\text{vrai positif}}{\text{vrai positif} + \text{faux négatif}}$$

Donne le pourcentage d'individus pouvant bénéficier d'un crédit correctement identifié par le modèle.

Spécificité :

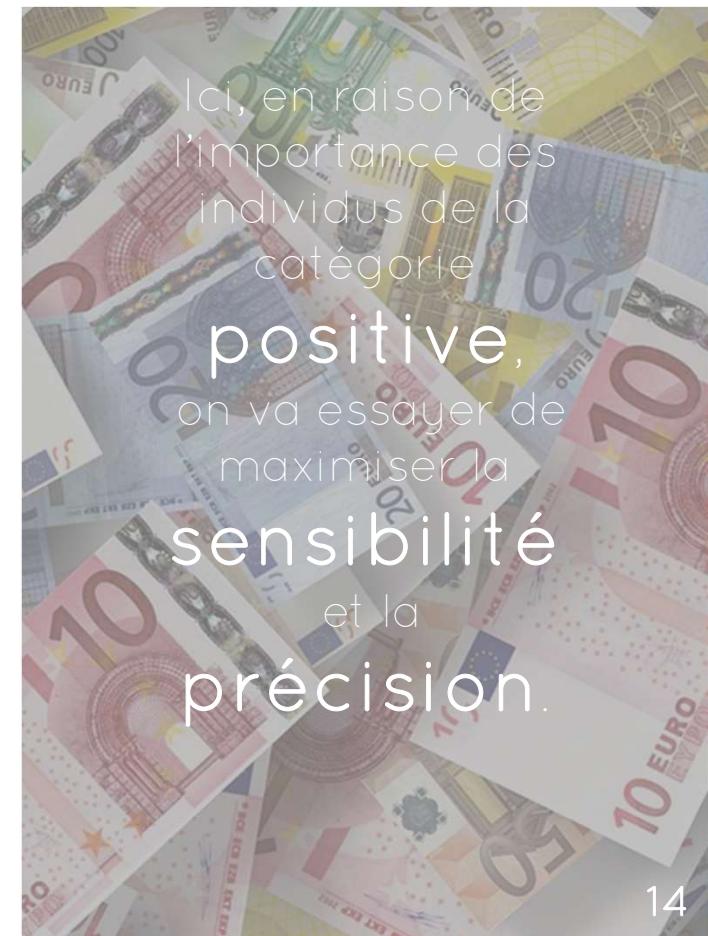
$$\frac{\text{vrai négatif}}{\text{vrai négatif} + \text{faux positif}}$$

Donne le pourcentage d'individus ne pouvant pas bénéficier d'un crédit correctement identifié par le modèle

Précision :

$$\frac{\text{vrai positif}}{\text{vrai positif} + \text{faux positif}}$$

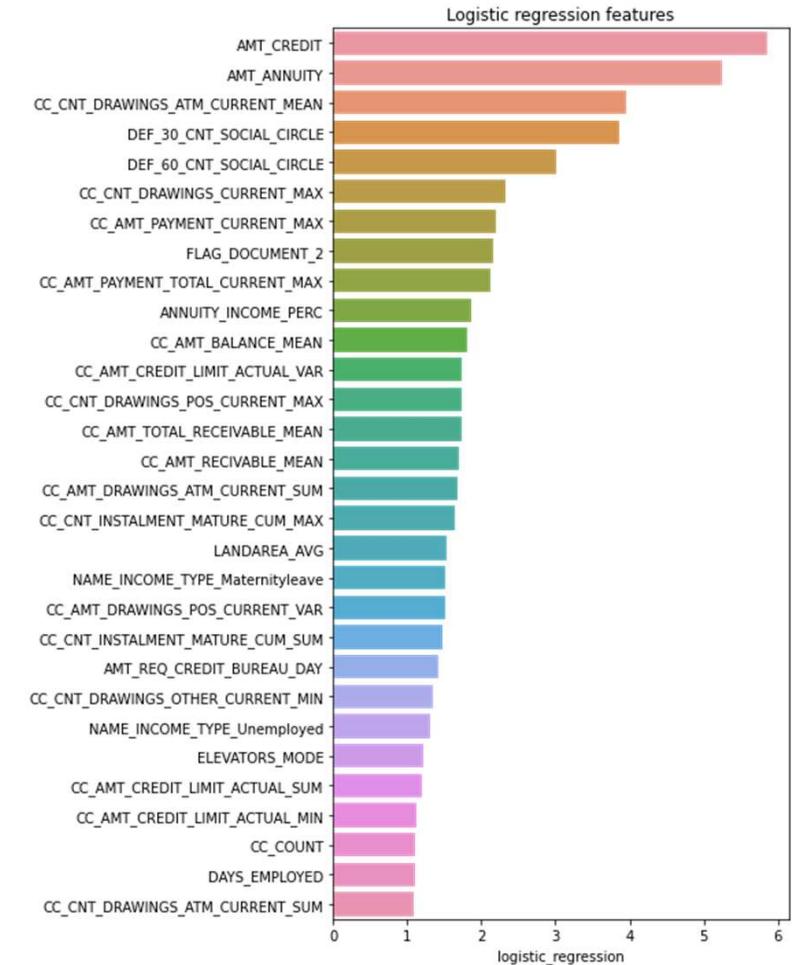
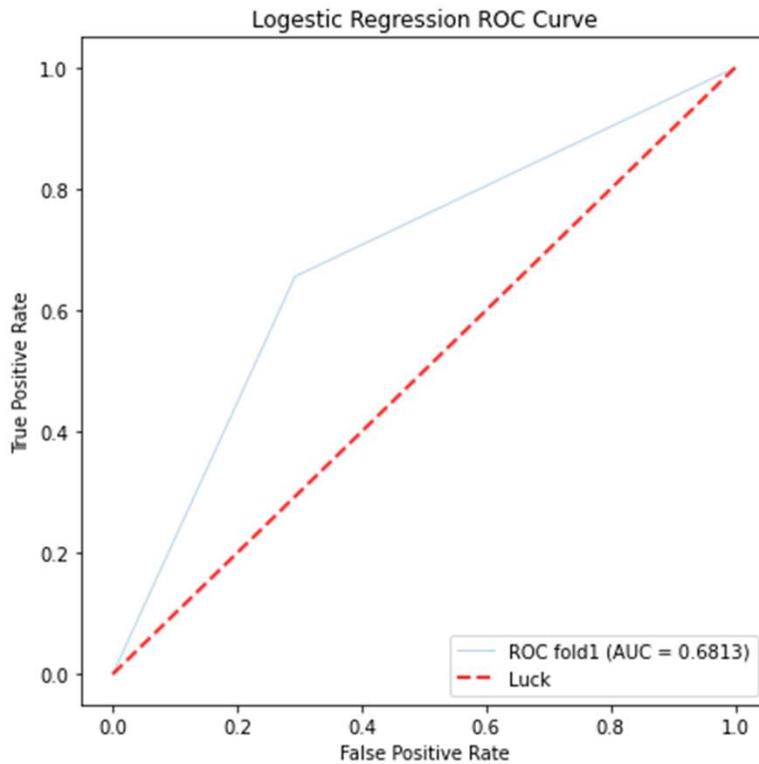
Donne le pourcentage de pouvant effectivement bénéficier d'un crédit par rapport aux personnes au nombre de personnes prédites comme telle.



II, RECHERCHE D'UN MODÈLE DE PRÉDICTION

b, Recherche d'un algorithme pertinent

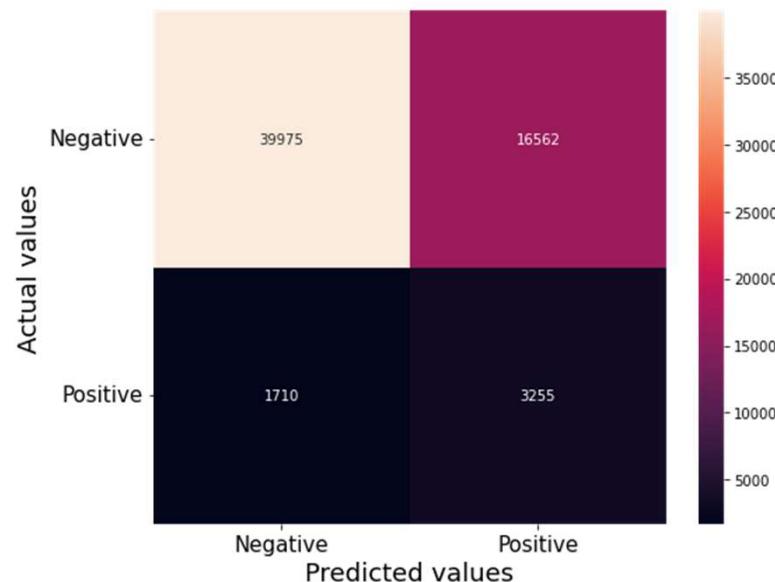
Régression logistique



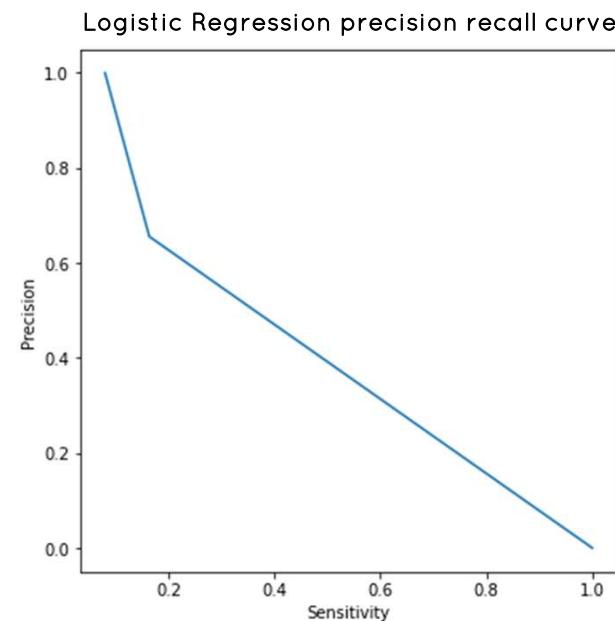
II, RECHERCHE D'UN MODÈLE DE PRÉDICTION

b, Recherche d'un algorithme pertinent

Régression logistique



score perso : -0,32
sensibilité : 65,56%
précision : 16,43%

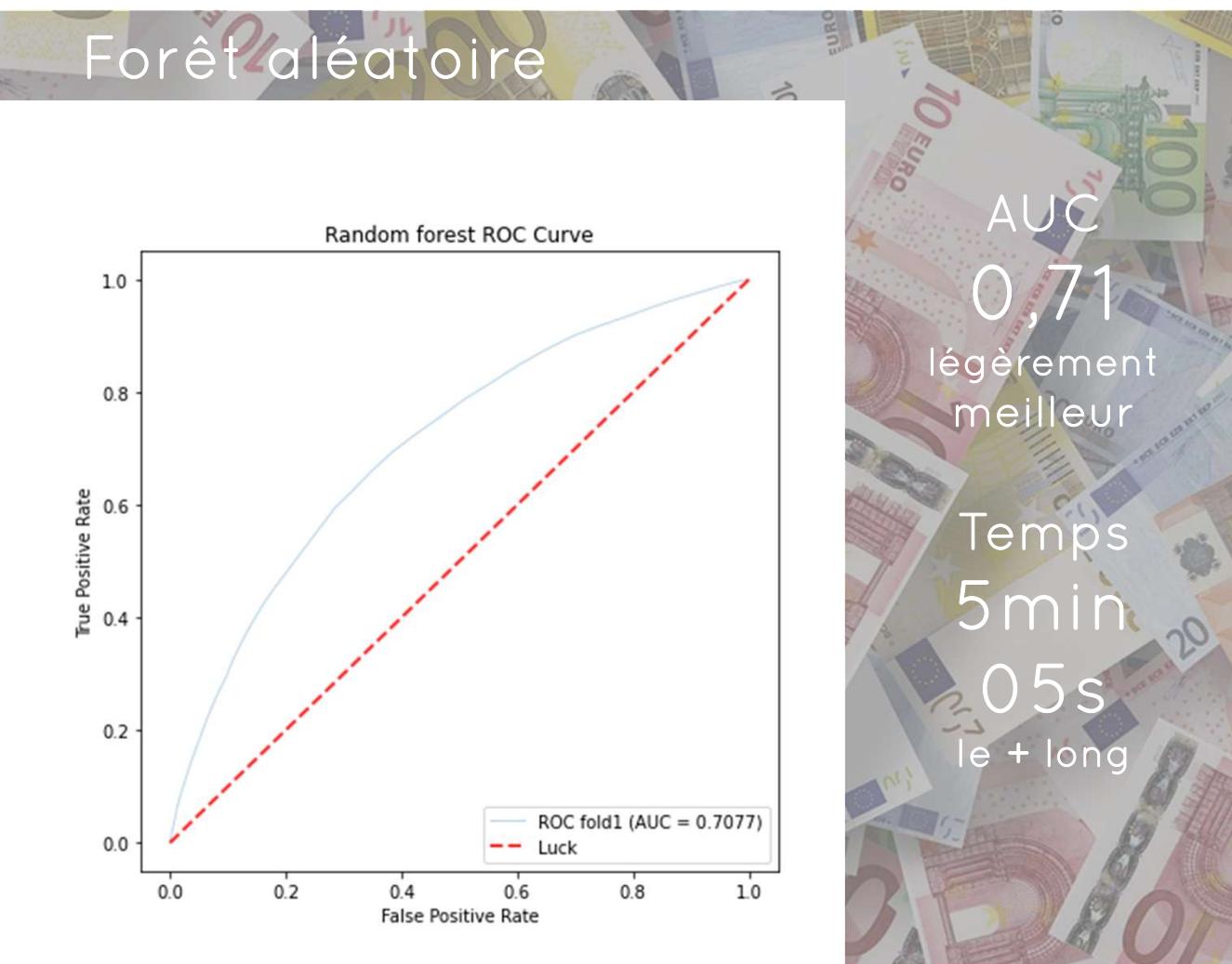


8/10 positif est un faux.

La régression logistique ne parvient pas à converger, cela démontre la **non linéarité** de la relation entre les données et la cible et explique les faibles scores obtenus.

II, RECHERCHE D'UN MODÈLE DE PRÉDICTION

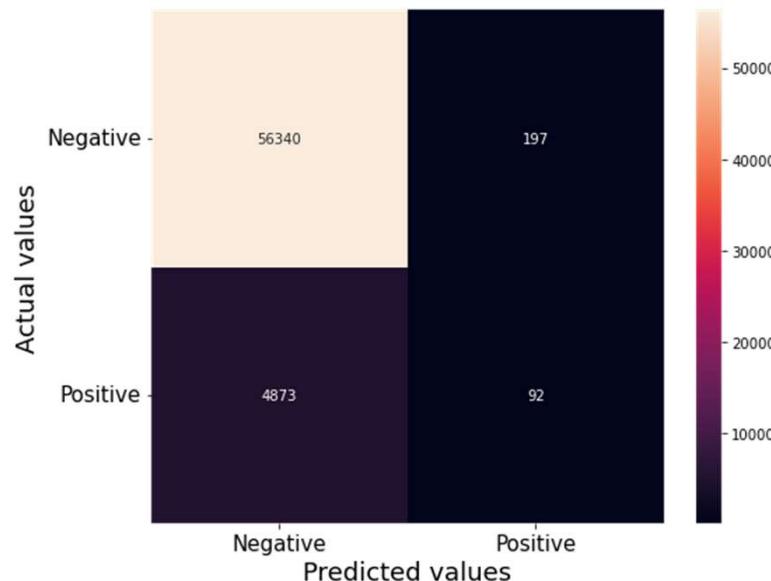
b, Recherche d'un algorithme pertinent



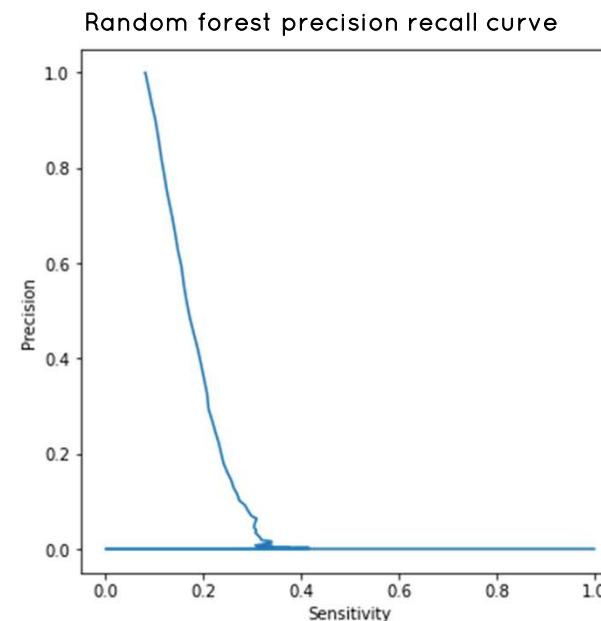
II, RECHERCHE D'UN MODÈLE DE PRÉDICTION

b, Recherche d'un algorithme pertinent

Forêt aléatoire



score perso : 0,91
sensibilité : 1,85%
précision : 31,83%



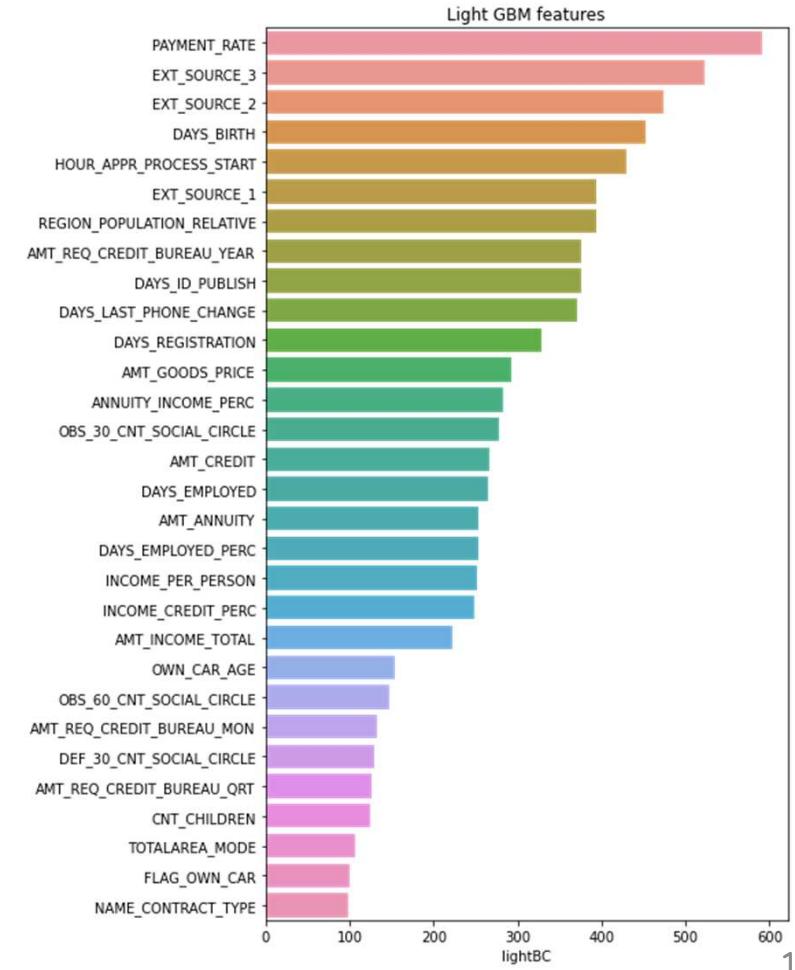
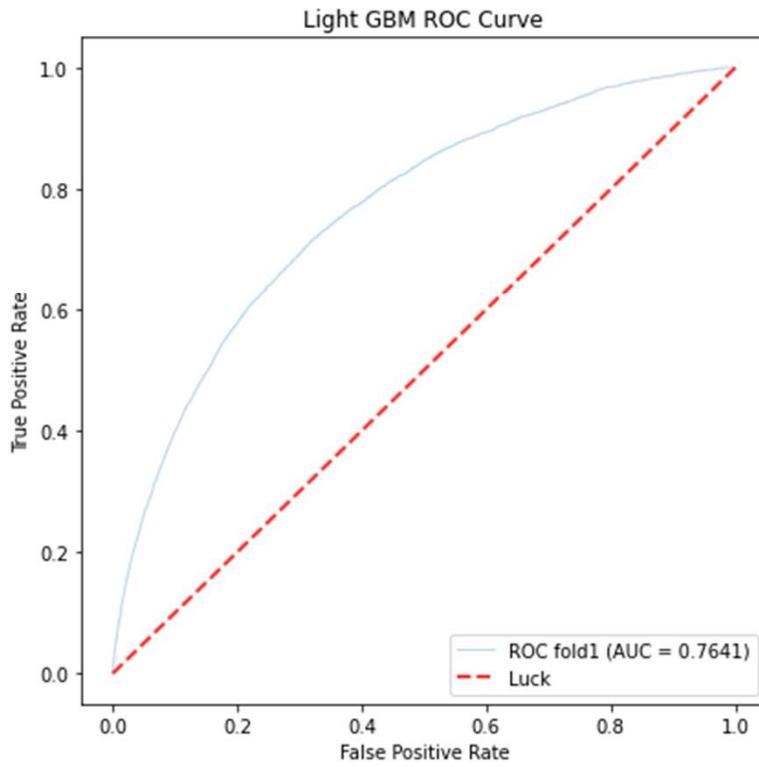
Bon score perso mais sensibilité et précision très mauvaises.

Moins de 2% de clients positifs sont identifiés.

II, RECHERCHE D'UN MODÈLE DE PRÉDICTION

b, Recherche d'un algorithme pertinent

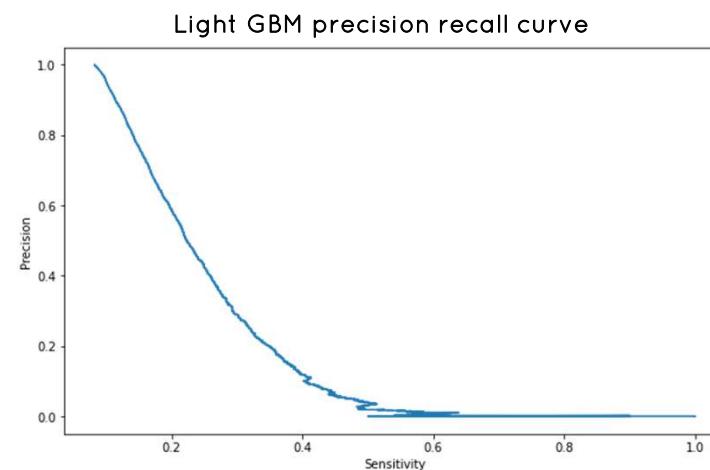
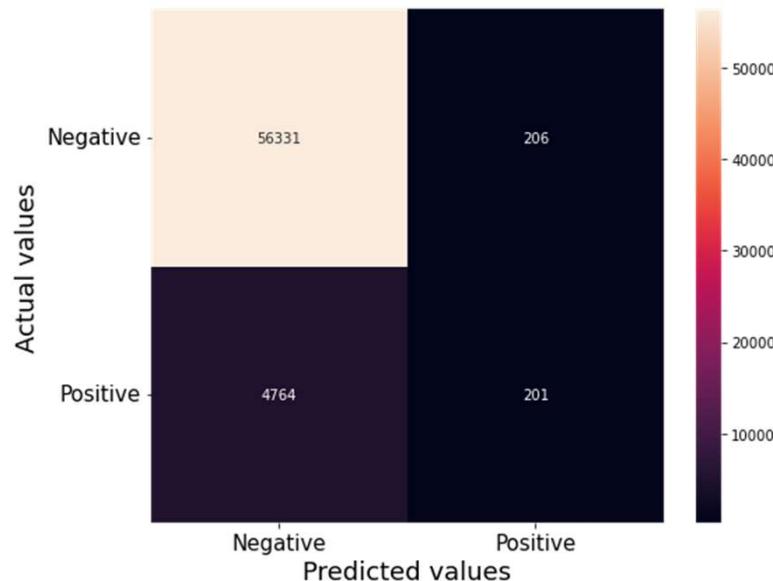
Light GBM



II, RECHERCHE D'UN MODÈLE DE PRÉDICTION

b, Recherche d'un algorithme pertinent

Light GBM



score perso : 0,91
sensibilité : 4,05%
précision : 49,39%

Score perso similaire à random forest.
Mais sensibilité et précision légèrement meilleures.

½ positif est un faux.

Encore insuffisant.

Des trois modèles, le **Light GBM** est le meilleur.

Mais les prédictions qu'il établi sont encore d'une **qualité trop faible** pour prétendre entrer en production en l'état.

Nous allons voir si nous pouvons l'améliorer avec une
recherche d'hyperparamètres
et celle d'un
seuil pertinent.

II, RECHERCHE D'UN MODÈLE DE PRÉDICTION

c, Optimisation de l'algorithme



Halving search CV

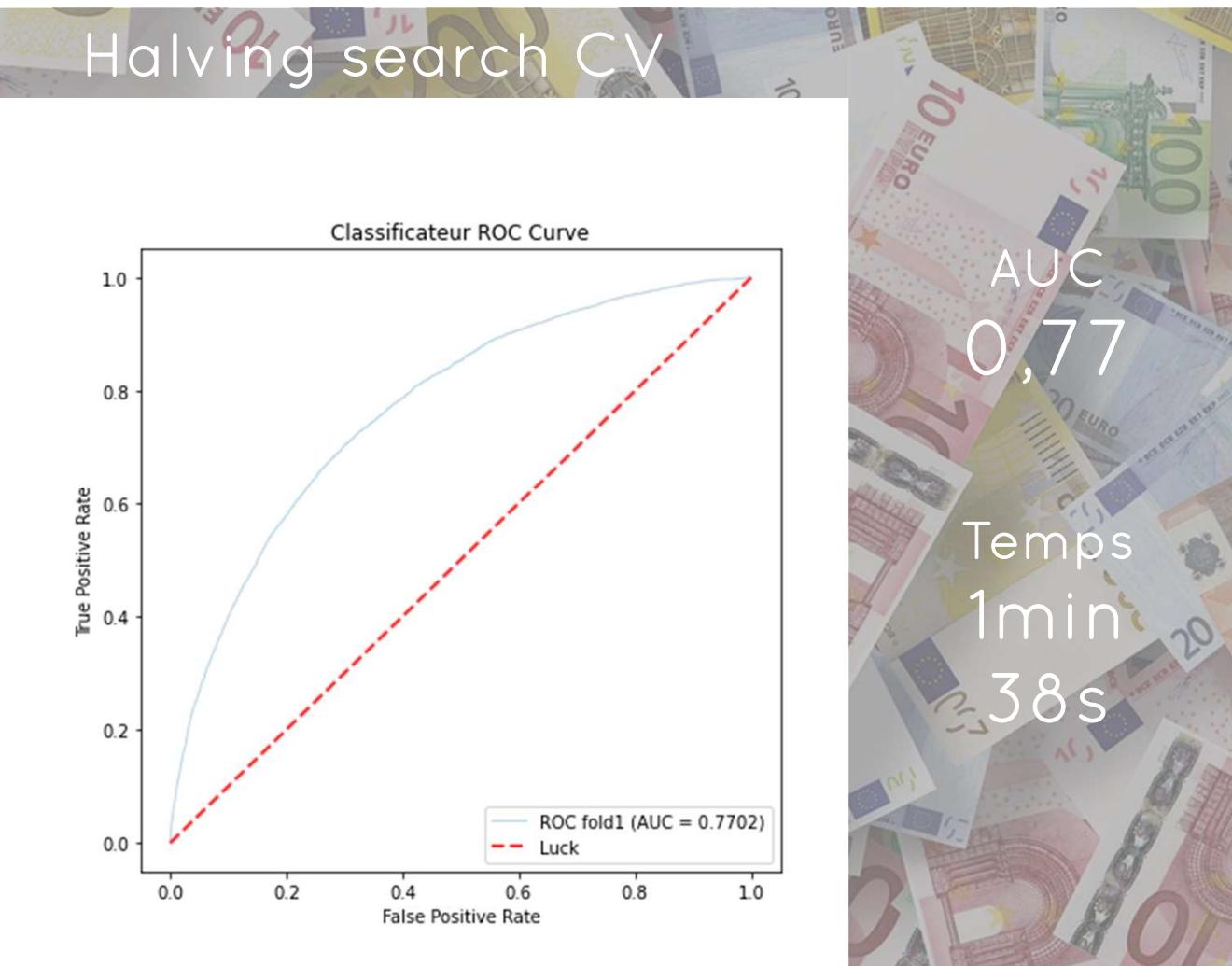
A la différence de GridSearchCV, **HalvingSearchCV** travail par round sur des échantillons de plus en plus important du jeu de données.

Comme dans notre cas nous avons un **jeu de donné de plus de 1GB**, cette solution apparaît comme plus économique en **énergie et puissance de calcul**.

Nous ne pouvons pas rechercher les valeurs optimales de tous les paramètres, nous choisissons de nous concentrer sur **la vitesse d'apprentissage, le nombre de feuilles maximum par arbres et le nombre d'arbres**.

II, RECHERCHE D'UN MODÈLE DE PRÉDICTION

c, Optimisation de l'algorithme



II, RECHERCHE D'UN MODÈLE DE PRÉDICTION

c, Optimisation de l'algorithme

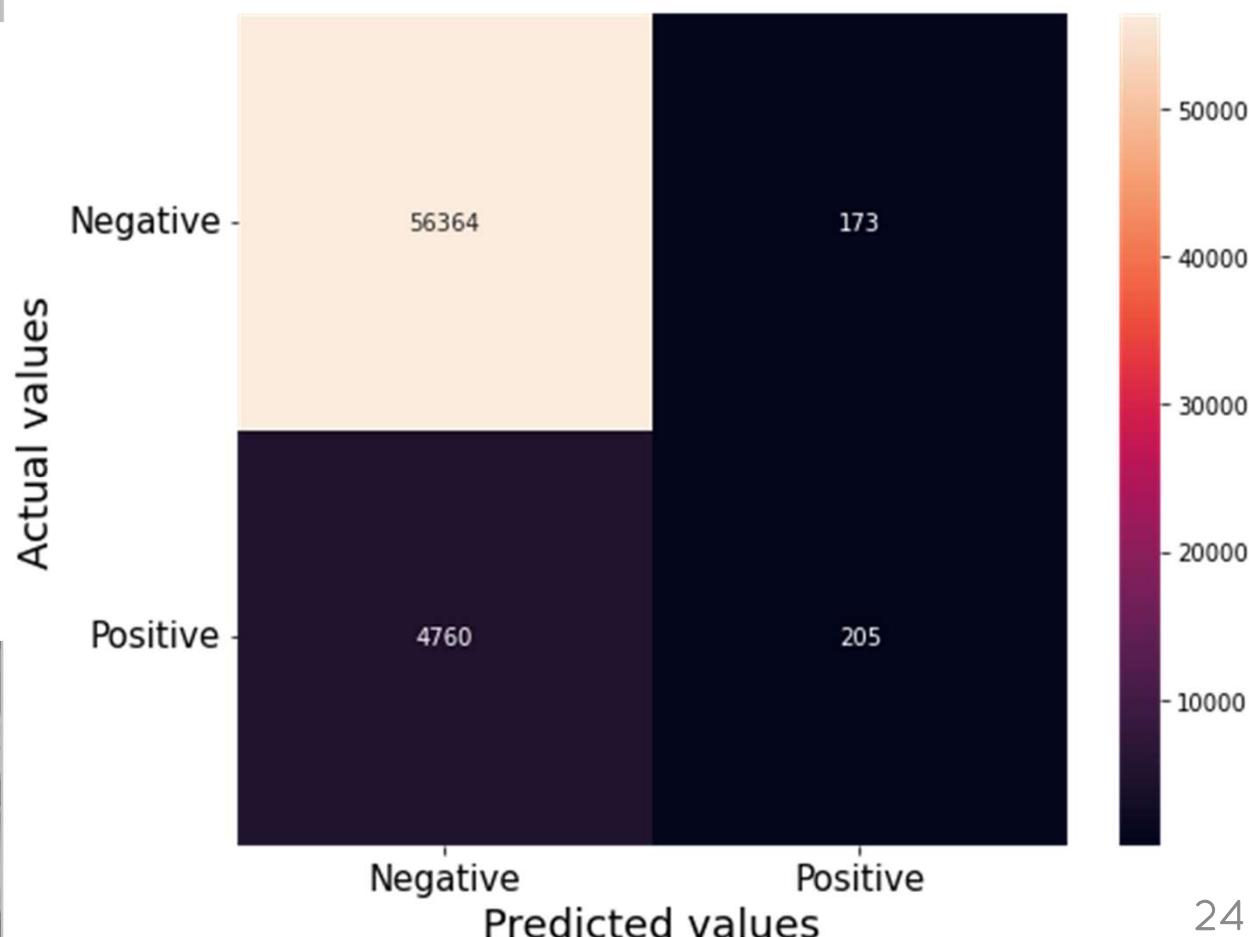
Halving search CV

La recherche d'hyperparamètres a permis un **légère amélioration** des prédictions du modèle.

Le score perso et la précision progressent.

La sensibilité est encore insuffisante.

score perso : 0,93
sensibilité : 4,13%
précision : 54,23%



II, RECHERCHE D'UN MODÈLE DE PRÉDICTION

c, Optimisation de l'algorithme

Recherche de Seuil

Modifier le seuil peut parfois permettre d'améliorer les performances du modèle.

Ici, on voit qu'un seuil à **0,4 et 0,5** produisent de très bons résultats.

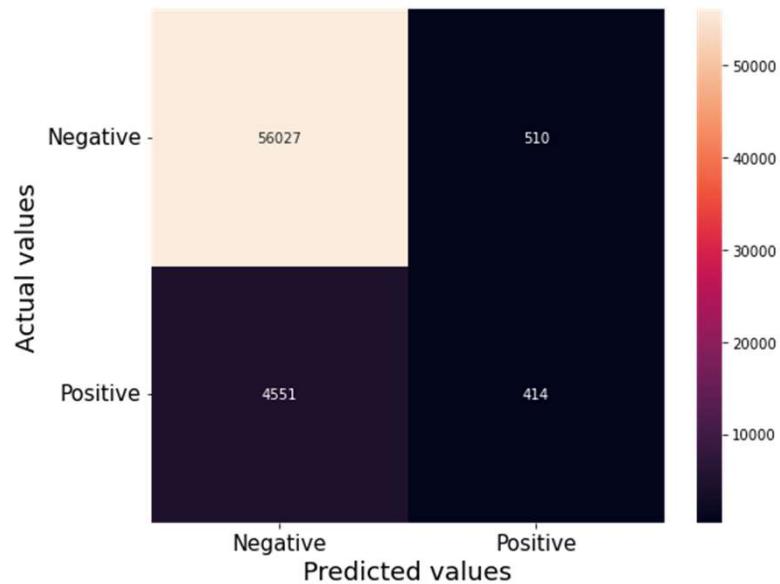
Les deux seuil peuvent être retenus en fonction de ce que l'entreprise veut privilégier

Seuil	Sensibilité	Précision	Score perso
0,1	59,1%	19,7%	-0,51
0,2	32,31%	29,3%	0,58
0,3	16,9%	36,8%	0,84
0,4	8,3%	44,8%	0,91
0,5	4,1%	54,2%	0,93
0,6	1,7%	63,8%	0,93
0,7	0,6%	69,1%	0,92
0,8	0,1%	83,3%	0,92
0,9	0,0%	100,0%	0,92

II, RECHERCHE D'UN MODÈLE DE PRÉDICTION

c, Optimisation de l'algorithme

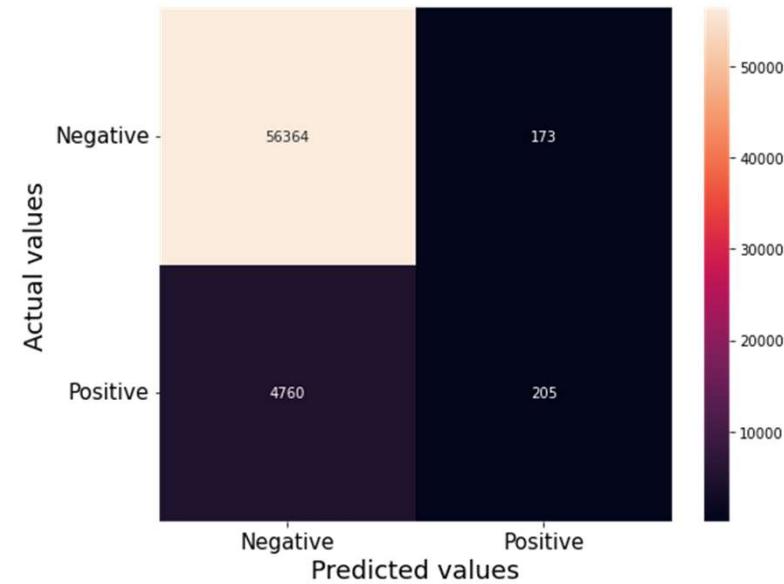
Recherche de Seuil



Seuil : 0,4

Privilégie la **sensibilité** (8%) à la précision (45%).

Plus de clients vraiment positifs sont détectés par le modèle



Seuil : 0,5

Maximise la **précision** (54%) par rapport à la sensibilité (4%).
Plus de certitude sur les clients identifiés comme positifs.

Après cette phase d'optimisation, on se retrouve avec un modèle pouvant entrer en production.

Cependant, la compagnie souhaite que les prédictions du modèle soit explicable aux clients.

Pour ce faire, nous allons devoir déterminer le poids de chaque critère dans la décision du modèle.

II, RECHERCHE D'UN MODÈLE DE PRÉDICTION

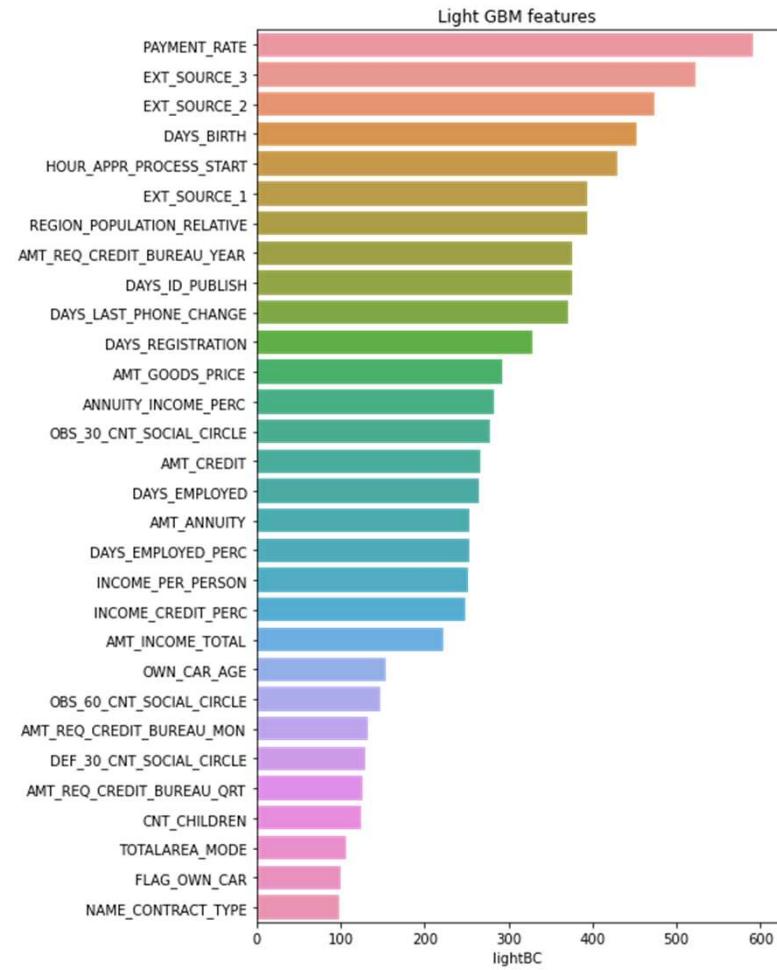
d, Analyse de l'importance des features

Les critères globaux

Ce sont ceux qui s'intéressent aux poids des critères pour l'ensemble des individus.

On les voit ici classés par ordre d'importance pour notre modèle.

On note l'importance des sources extérieures qui malheureusement sont plutôt obscures.



II, RECHERCHE D'UN MODÈLE DE PRÉDICTION

d, Analyse de l'importance des features

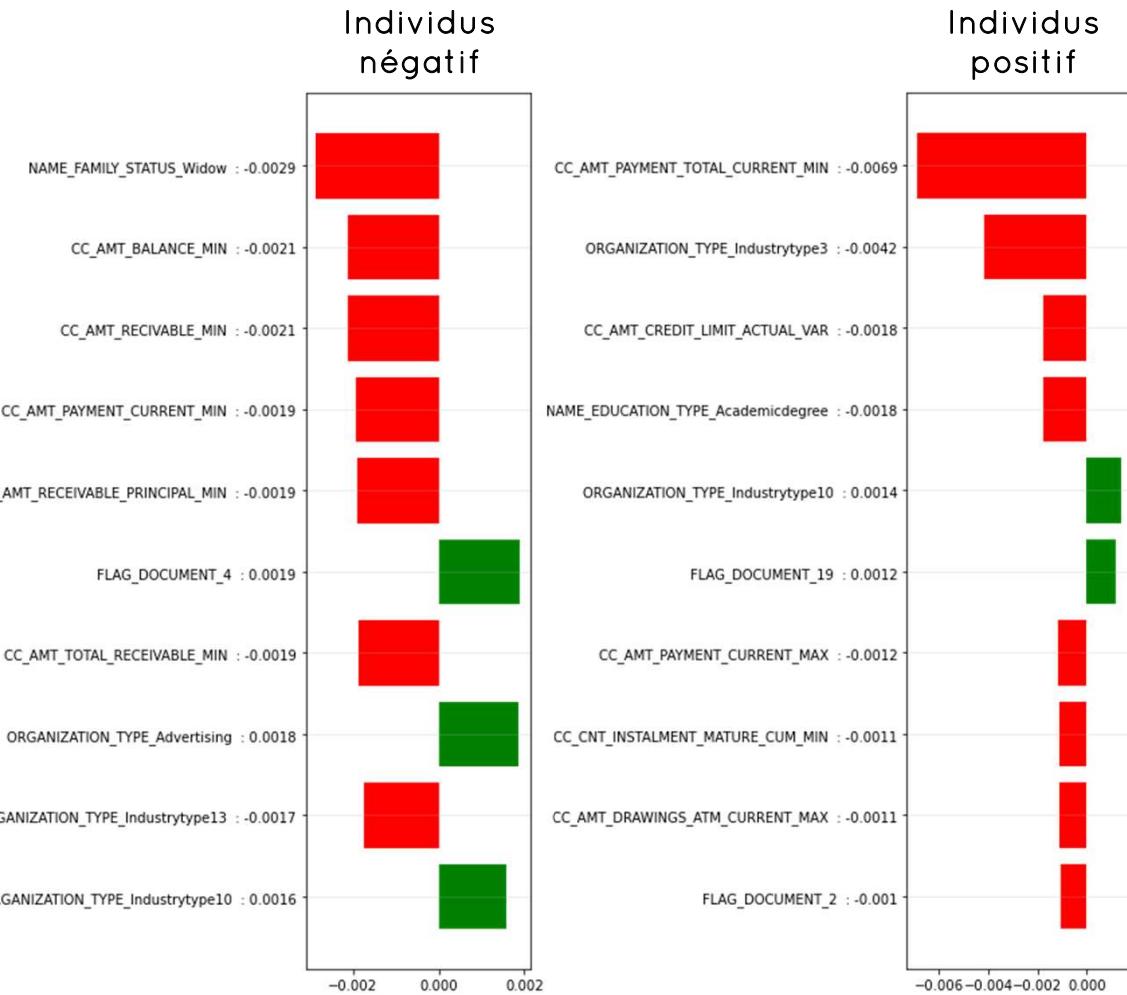
Les critères locaux

Avec LIME

(Local Interpretable Model-agnostic
Explanations)

 Permet une lecture facile du
poids des critères.
Méthode **rapide et économique** en
calcul.

 Ne donne pas d'informations
sur le **score** total de l'individus.



II, RECHERCHE D'UN MODÈLE DE PRÉDICTION

d, Analyse de l'importance des features

Les critères locaux

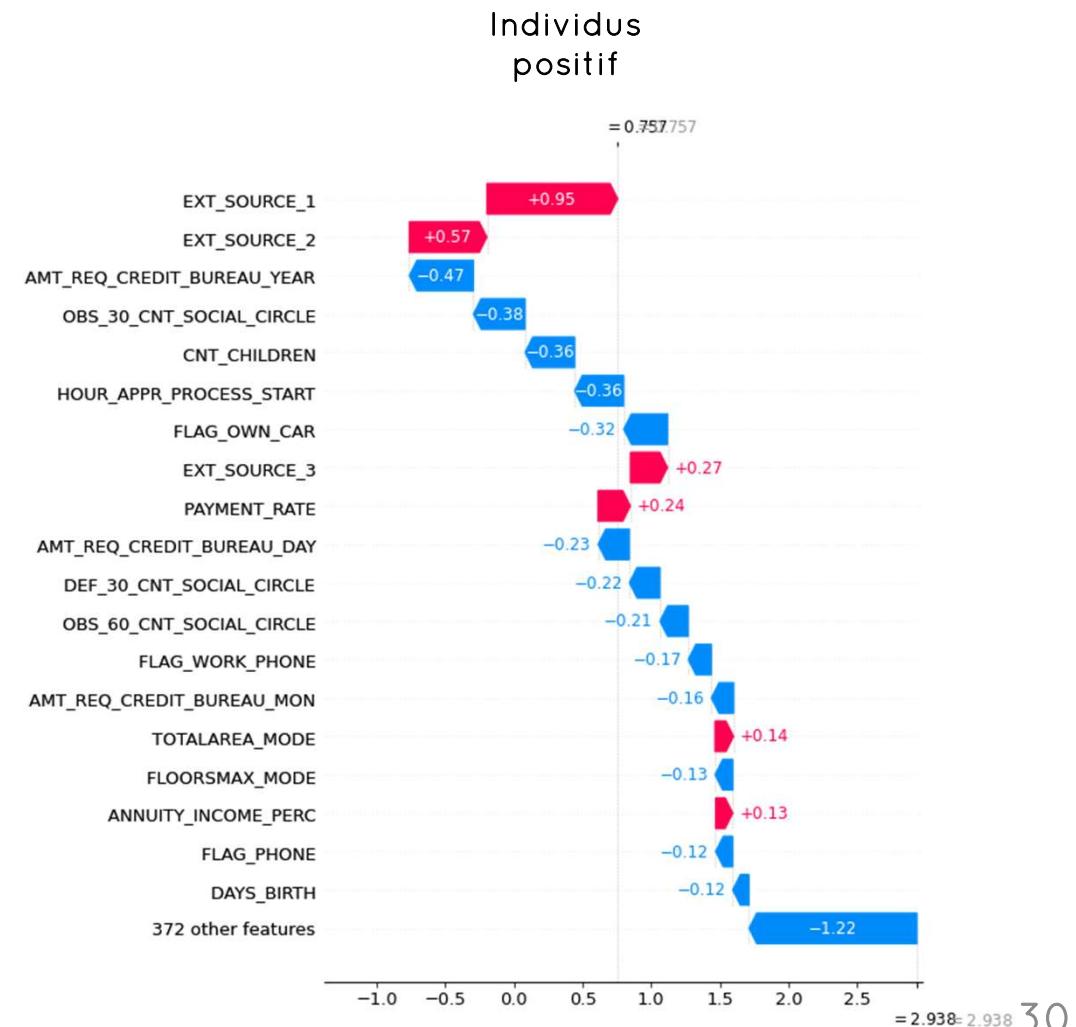
Avec SHAP

(Shapley Additive exPlanations)

 Donne un **score** à l'individus, ici 0,757. Si ce score est positif, l'individus voit son crédit accepté.

Montre quels critères ont le plus de poids.

 Méthode demandant beaucoup de **puissance de calcul** car elle calcule l'ensemble des individus.



II, RECHERCHE D'UN MODÈLE DE PRÉDICTION

d, Analyse de l'importance des features

Exemple d'un individu négatif

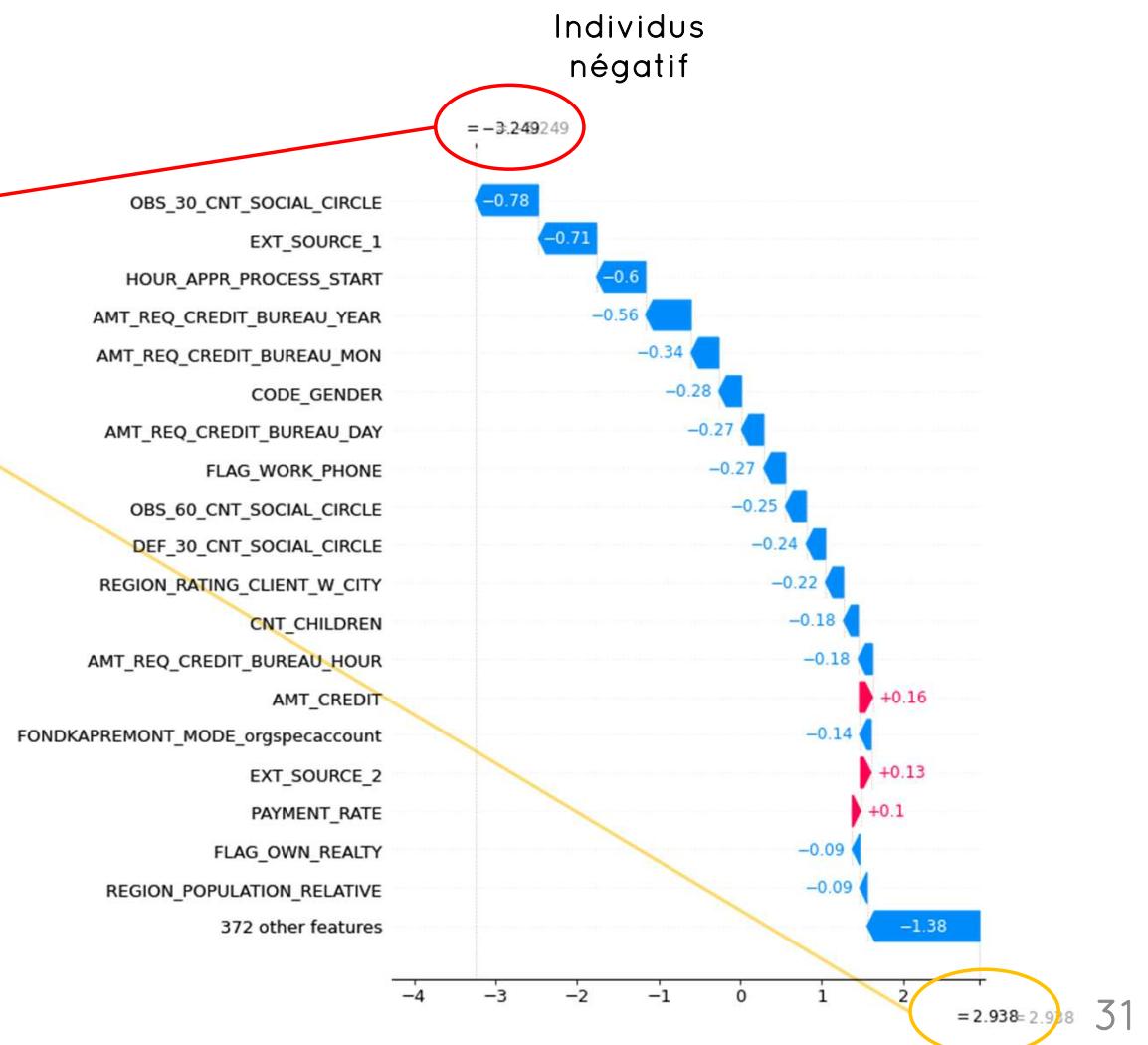
score de l'individu

score maximal

Ici le score de l'individus est de -3,249.

Si il n'avait pas de pénalité sur les 6 critères les plus importants, il serait positif.

SHAP permet également de donner des conseils pour améliorer un score.



Nous avons désormais tous les éléments pour la mise en production.

Pour partager notre modèle, nous devons **l'exporter**.

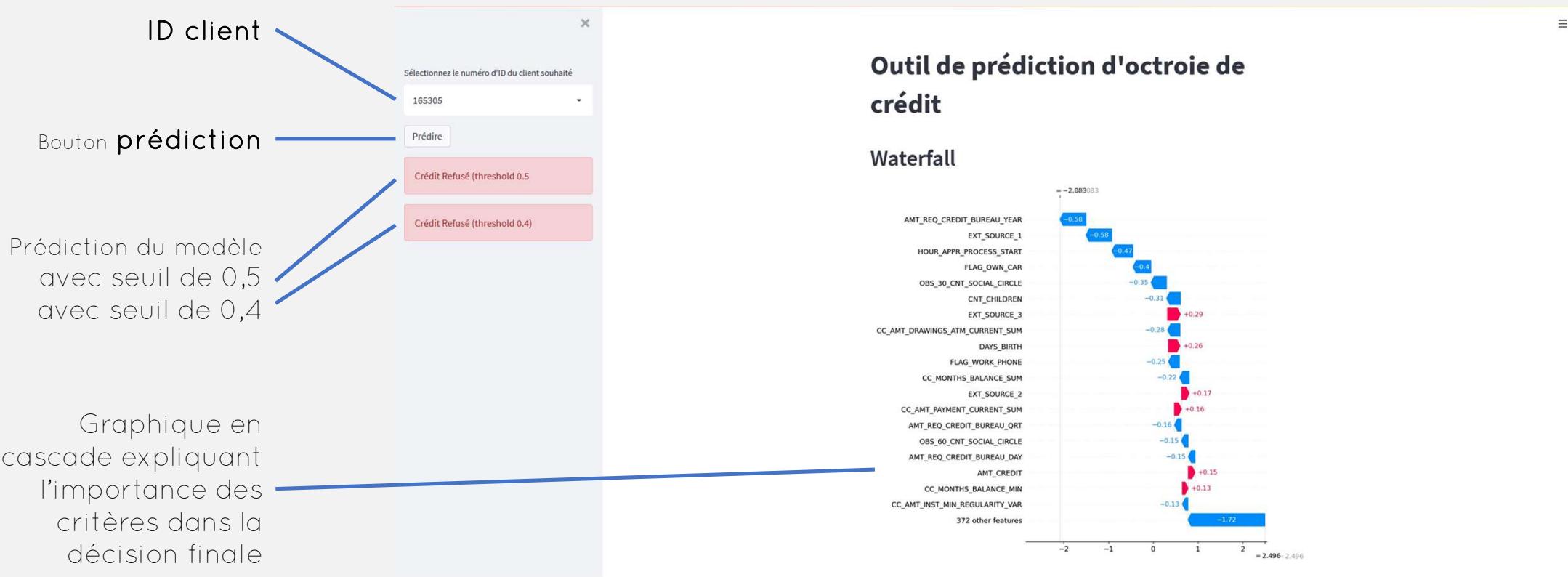
Les éléments que nous exportons sont :

- le **pipeline** du modèle contenant la création des individus positifs supplémentaires avec **SMOTE** et l'**algorithme entraîné avec les meilleurs paramètres** trouvés
- l'algorithme entraîné avec **SHAP** pour expliquer les critères locaux par rapport à l'ensemble

L'entraînement de ces deux éléments demande de la puissance de **calcul et du temps**, le fait de les exporter déjà entraîné permet de **fluidifier** notre application future.

III, L'outils final : le Dashboard

a, L'exemple d'un client négatif

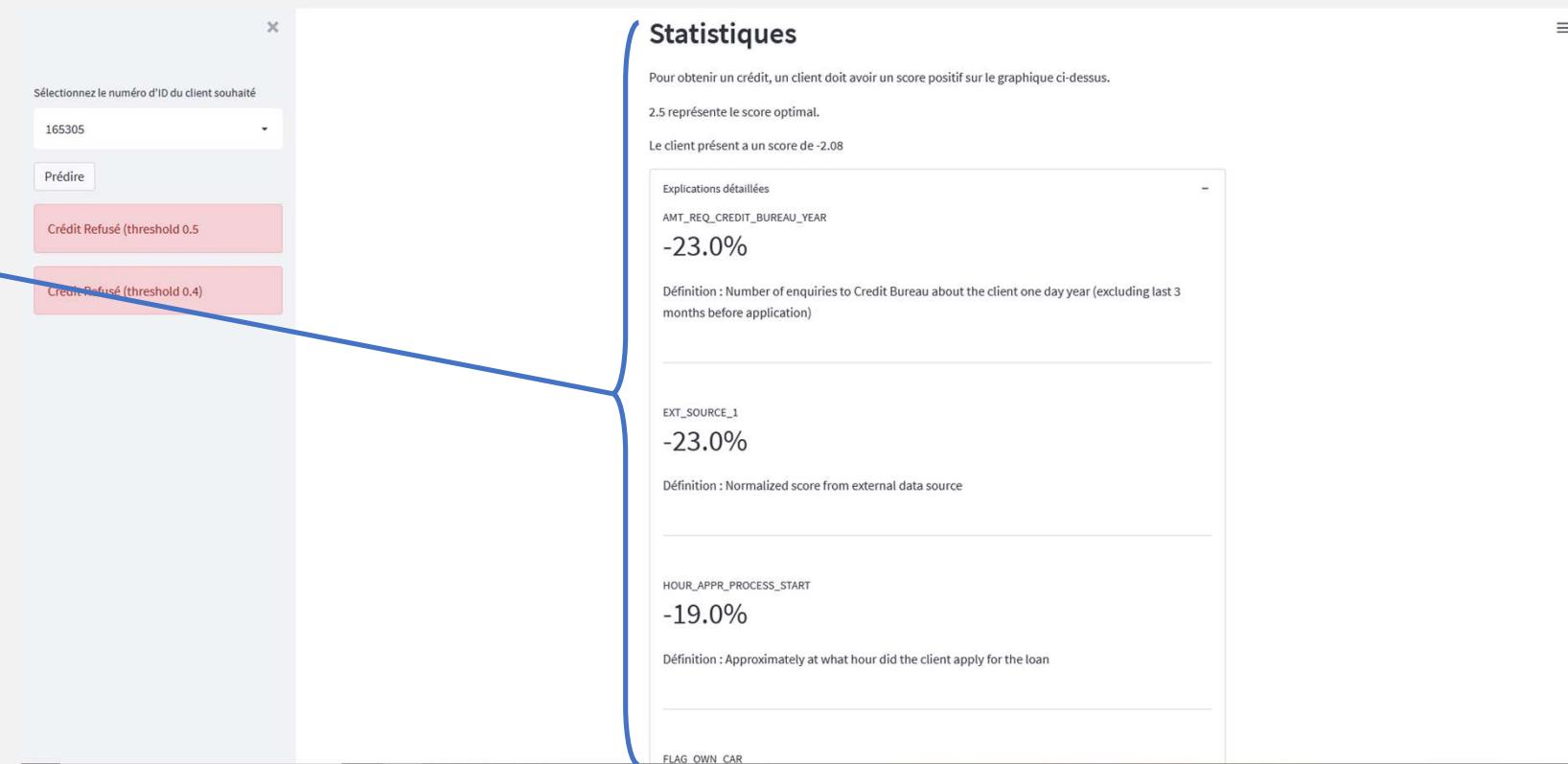


III, L'outils final : le Dashboard

a, L'exemple d'un client négatif

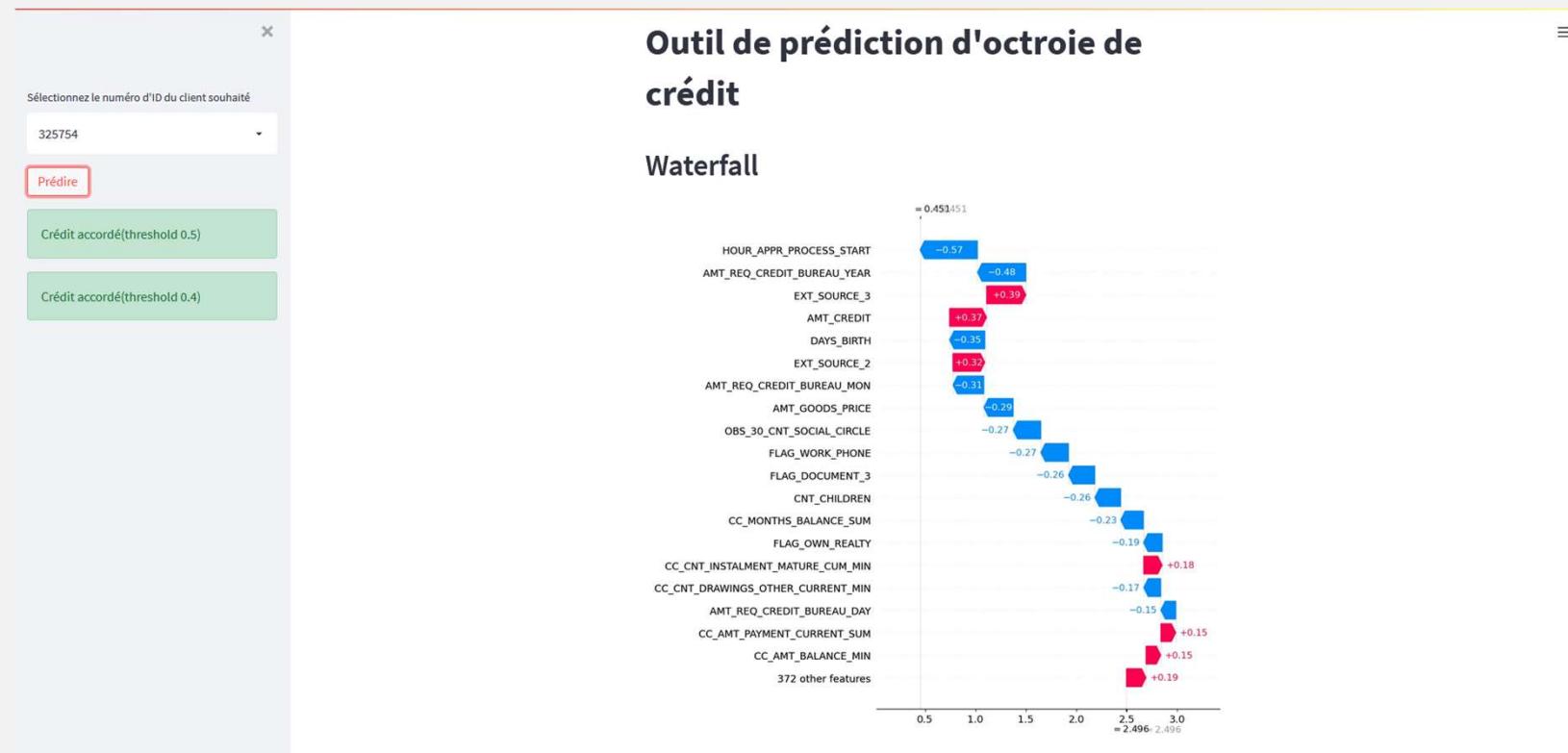
Les statistiques du client en %

Par exemple, ici, le nombre de demande du client au Bureau de crédit sur l'année écoulée, l'a pénalisé à hauteur de 28% dans sa demande.



III, L'outils final : le Dashboard

b, L'exemple d'un client positif



III, L'outils final : le Dashboard

b, L'exemple d'un client positif

Sélectionnez le numéro d'ID du client souhaité

Prédire

Crédit accordé(threshold 0.5)

Crédit accordé(threshold 0.4)

Statistiques

Pour obtenir un crédit, un client doit avoir un score positif sur le graphique ci-dessus.

2.5 représente le score optimal.

Le client présent a un score de 0.45

Explications détaillées
HOUR_APPR_PROCESS_START -23.0% Définition : Approximately at what hour did the client apply for the loan
AMT_REQ_CREDIT_BUREAU_YEAR -19.0% Définition : Number of enquiries to Credit Bureau about the client one day year (excluding last 3 months before application)
EXT_SOURCE_3 16.0% Définition : Normalized score from external data source
AMT_CREDIT