Projet 7 : Implémentez un modèle de scoring

Problématique

Problématique : développer un dashboard interactif pour que les chargés de relation client puissent à la fois expliquer de façon la plus transparente possible les décisions d'octroi de crédits, mais également permettre à leurs clients de disposer de leurs informations personnelles et de les explorer facilement

Problématique

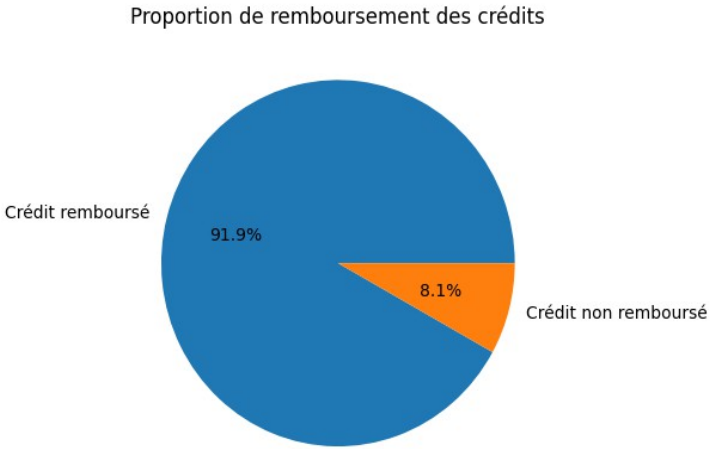
- L'entreprise « Prêt à dépenser » souhaite mettre en œuvre un outil de « scoring crédit » pour calculer la probabilité qu'un client rembourse son crédit, puis classifie la demande en crédit accordée ou refusée
- Elle souhaite donc développer un algorithme de classification et pouvoir présenter les résultats de manière lisible pour que les chargés de clientèle puissent faire un retour au client, dans une optique de transparence
- Elle souhaite en particulier éviter les clients qui ne pourront pas rembourser leur crédit

Présentation des données

- On a 10 dataframes contenant des informations relatives aux clients
- Un dataframe principal contenant des informations connues de la banque, indiquant si le client est capable de rembourser son crédit
- 9 dataframes issus de source tierce, contenant des informations complémentaires, sur le client, comme ses crédits précédents, ses informations bancaires …
- On rassemble toutes ses informations dans un seul dataframe pour avoir une source unique de données

Modélisation

- On commence par regarder la proportion de clients ayant remboursé leur crédit

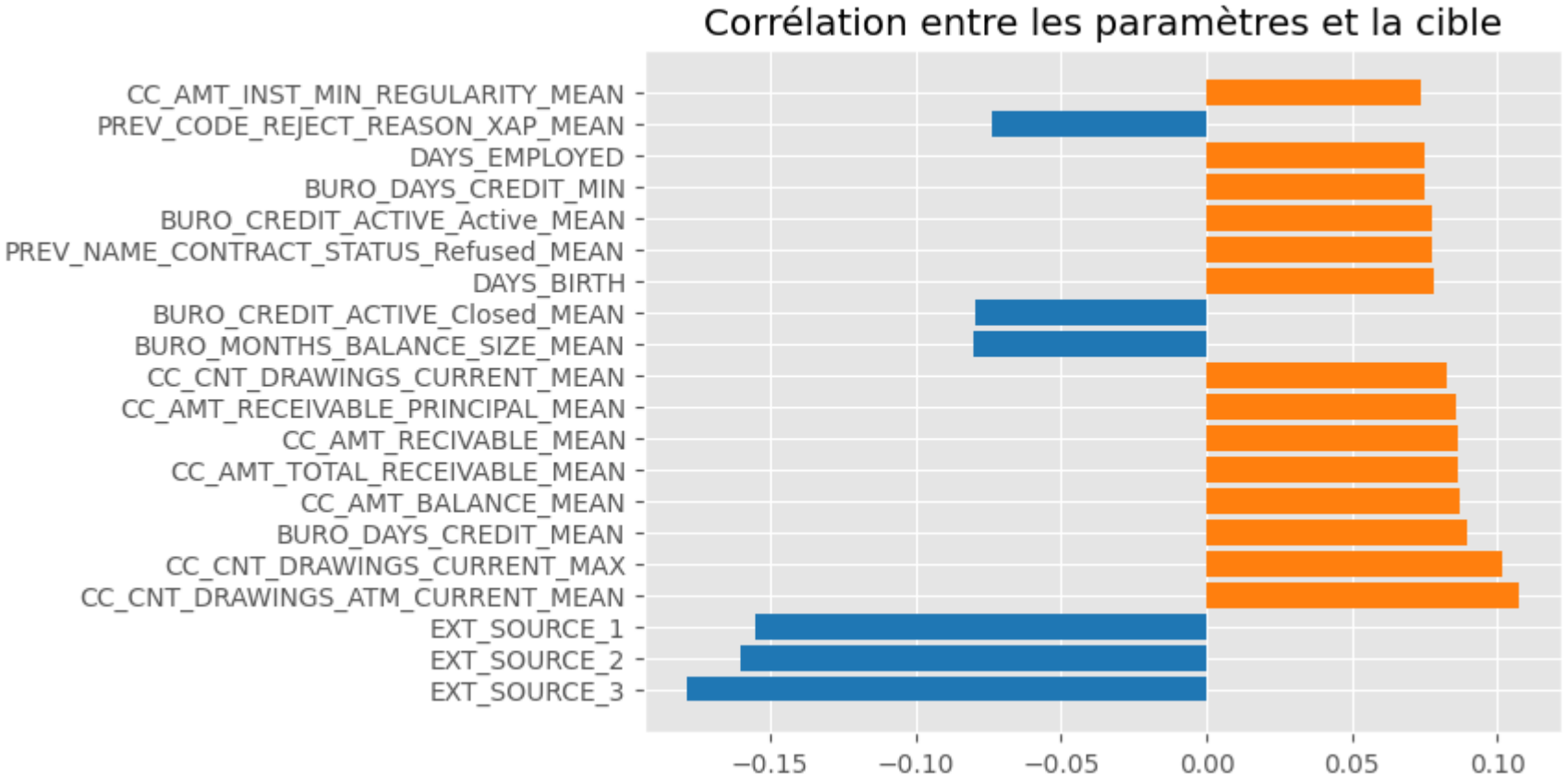


- On constate qu'il y a beaucoup plus de clients ayant remboursé leur crédit que de client mauvais payeur

Modélisation

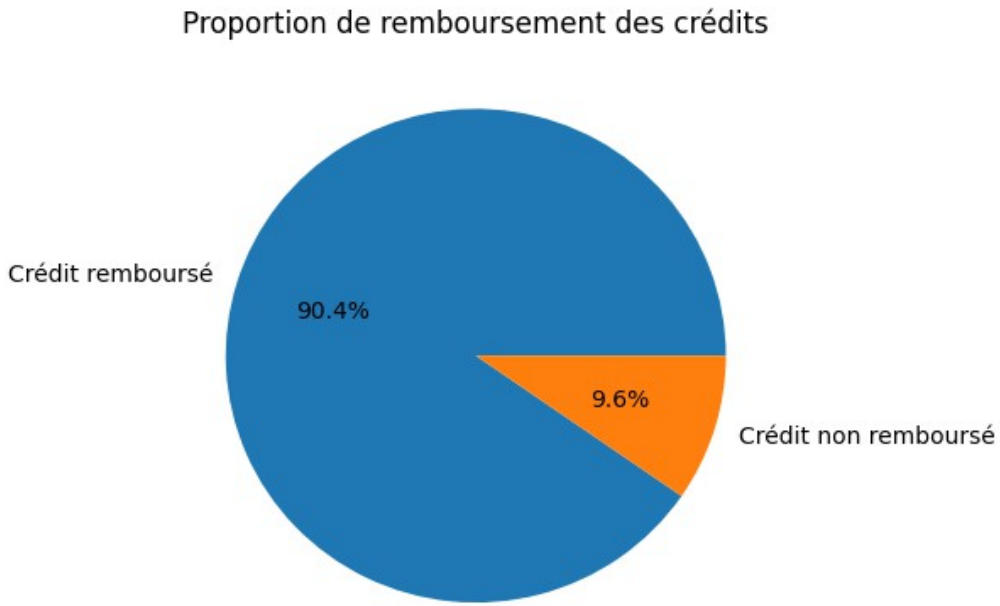
- On sélectionne les 20 caractéristiques ayant la plus grande corrélation avec notre cible
- On utilise la méthode des corrélations pour sélectionner ces caractéristiques
- On prend uniquement les clients ayant toutes ces informations renseignées

Modélisation



Modélisation

- On vérifie que l'on a toujours le même déséquilibre



Modélisation

- Si on laisse le déséquilibre, nos modèles risquent de prioriser la classe majoritaire
- Pour compenser le déséquilibre, on utilise SMOTE (Synthétic Minority Oversampling Technique)
- On intègre le SMOTE à un Pipeline
- L'évaluation des modèles se fera sur le véritable dataset d’entraînement (non équilibré)

Modélisation

- « Prêt à dépenser » veut que notre modèle soit très performant pour éviter les mauvais payeurs (faux négatifs)
- On évalue la performance de nos modèles avec un fbeta_score, qui pénalise les faux négatifs
- On entraîne nos modèles avec deux types de score : AUC ROC et fbeta_score
- On teste divers modèles de machine learning avec divers paramètres

Modélisation

	model	score	best_params	best_score	train	test	fbeta_score_test
0	search_tree	fbeta	{'classification__n_estimators': 13, 'classifi...	0.526590	0.527499	0.528562	0.528562
0	search_tree	roc_auc	{'classification__n_estimators': 13, 'classifi...	0.613108	0.611411	0.614265	0.487458
0	search_logi	fbeta	{'classification__tol': 0.0047508101621027985,...	0.642679	0.640188	0.625189	0.625189
0	search_logi	roc_auc	{'classification__tol': 0.0001668100537200059,...	0.609928	0.616308	0.591660	0.550495
0	search_KNN	fbeta	{'classification__n_neighbors': 10}	0.416634	0.932034	0.368335	0.368335
0	search_KNN	roc_auc	{'classification__n_neighbors': 10}	0.535191	0.831119	0.515405	0.375276
0	search_gradiant	fbeta	{'classification__n_estimators': 10, 'classifi...	0.492654	0.493365	0.499841	0.499841
0	search_gradiant	au_roc	{'classification__n_estimators': 43, 'classifi...	0.618447	0.623650	0.616262	0.457393

Modélisation

- La régression logistique donne les meilleurs résultats
- On affine notre modèle
- On fait un premier entraînement sur un échantillon avant de le faire sur l'ensemble du dataset

	model	score	best_params	best_score	train	test	fbeta_score_test
0	search_logi	fbeta	{'classification__tol': 0.0022570197196339213,...	0.626346	0.654305	0.662170	0.662170
0	search_logi	roc	{'classification__tol': 0.0016297508346206436,...	0.655095	0.664117	0.667156	0.630892

Modélisation

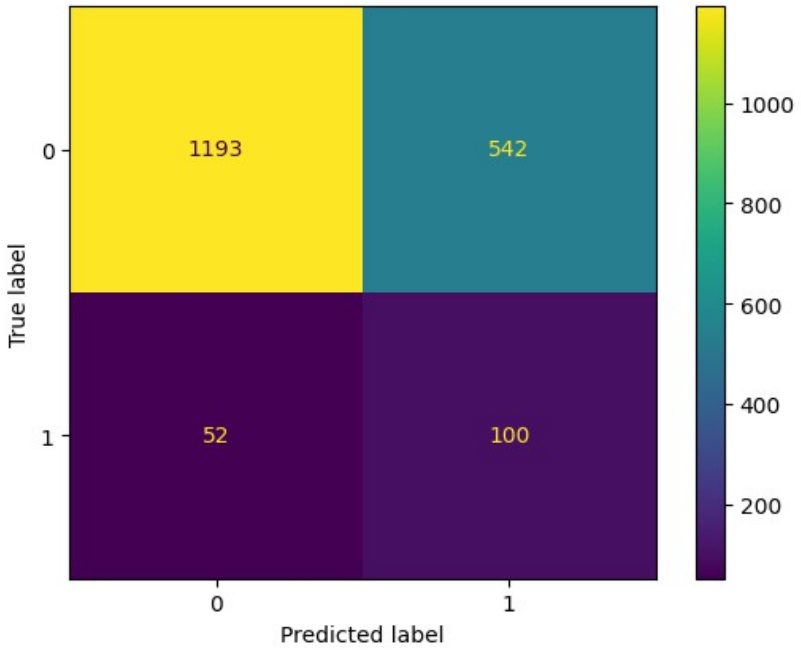
- On entraîne nos modèles sur l'ensemble de notre dataset d'entraînement

	model	score	best_params	best_score	train	test	fbeta_score_test
0	search_logi	fbeta	{'classification__tol': 0.0022570197196339213,...	0.637733	0.648908	0.63121	0.63121

	model	score	best_params	best_score	train	test	fbeta_score_test
0	search_logi	roc	{'classification__tol': 0.0016297508346206436,...	0.669061	0.670158	0.672751	0.637546

Modélisation

- On regarde la matrice de confusion de notre meilleur modèle :



Cela correspond à :

63.22% de vrais négatifs

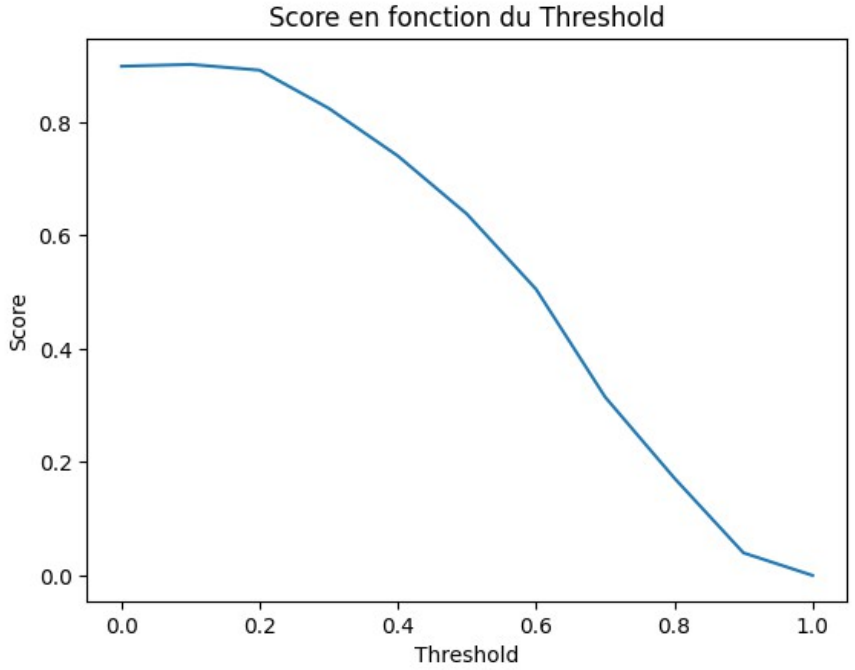
5.30% de vrais positifs

28.72% de faux positifs

2.76% de faux négatifs

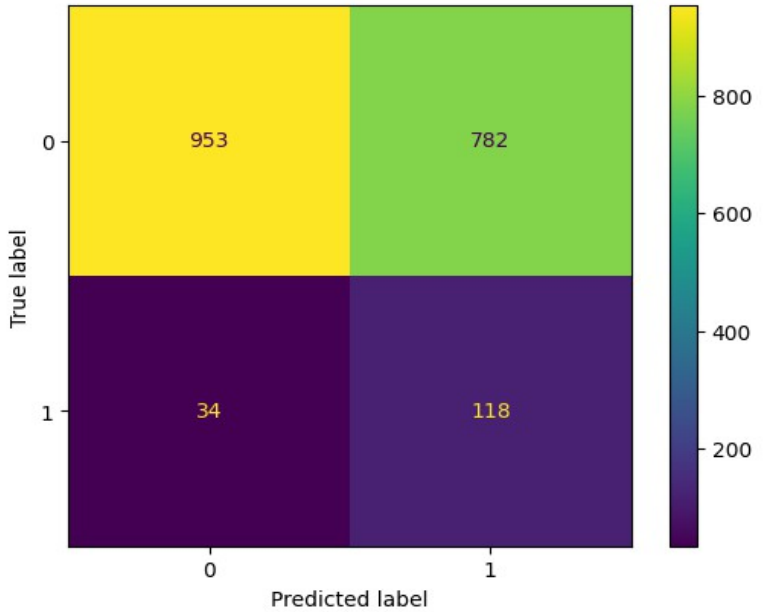
Modélisation

- On teste le modèle avec un nouveau seuil (threshold):



Modélisation

- On teste le modèle avec un seuil de 0.4



Cela correspond à :

50.50% de vrais négatifs

6.25% de vrais positifs

41.44% de faux positifs

1.80% de faux négatifs

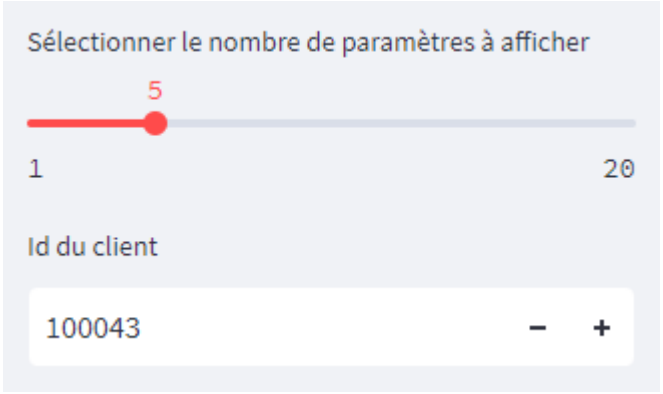
API

- Notre API va nous permettre d’appeler notre modèle et nous permettre d'avoir :
  - La prédiction
  - L'importance locale des paramètres
  - L'importance globale des paramètres
  - Les paramètres des clients ayant obtenu un crédit et celles de ceux n'en ayant pas obtenu
- L'API a été déployé via Heroku

Streamlit

- Une fois notre modèle mis en ligne avec une API pour obtenir diverses informations, il faut encore la rendre intelligible pour les clients et les chargés de relation
- On laisse le client choisir le nombre de paramètres qu'il veut afficher
- Les clients sont identifiés par leur numéro client dans l'interface

Streamlit


Sélectionner le nombre de paramètres à afficher
5
1
20
Id du client
100043

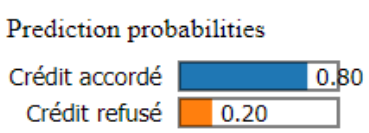
Streamlit

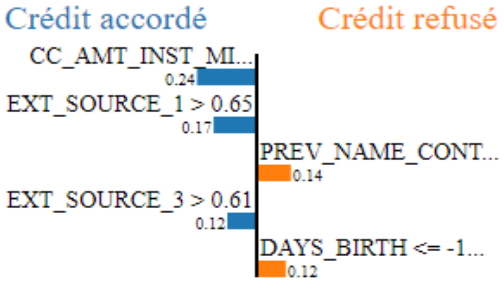
Demande de crédit :

Votre demande de crédit est :

Accordé !

Importance des caractéristiques locales



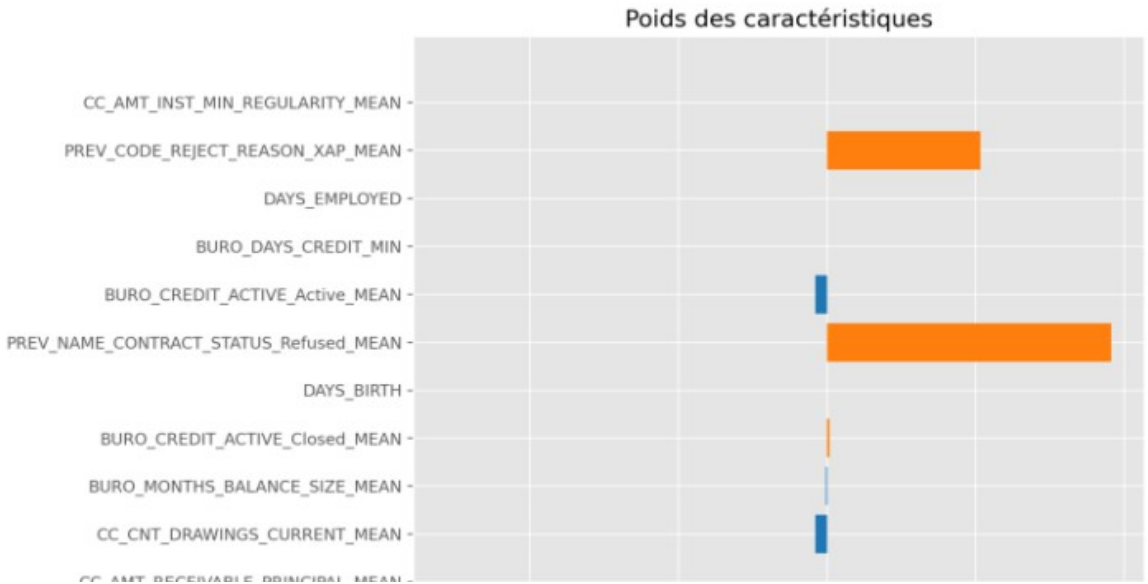


Feature	Value
CC_AMT_INST_MIN_REGULARITY_MEAN	11279.11
EXT_SOURCE_1	0.84
PREV_NAME_CONTRACT_STATUS_Refused_MEAN	0.56
EXT_SOURCE_3	0.75
DAYS_BIRTH	-17199.00

Streamlit

Importance des caractéristiques globales

Quel est le poid des caractéristiques dans le modèle utilisé.



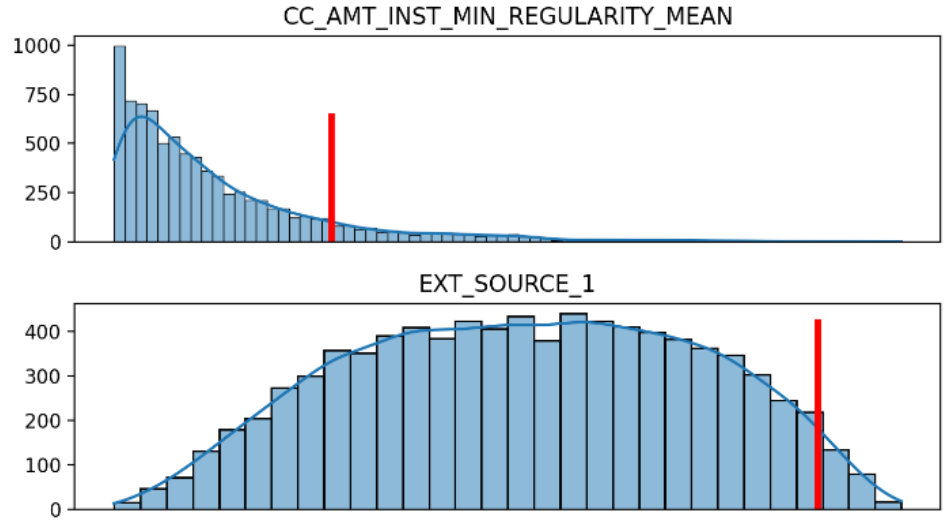
Streamlit

Répartition des caractéristiques

Comment vous situez-vous par rapport aux autres clients ?

Selectionner une comparaison :

- (●) Crédit accepté
- ( ) Crédit refusé



Streamlit

Répartition des caractéristiques

Comment vous situez-vous par rapport aux autres clients ?

Selectionner une comparaison :

- ○ Crédit accepté
- ◉ Crédit refusé