

Implémentez un modèle de scoring

Yonss JOSE

04/2022



Résumé du projet

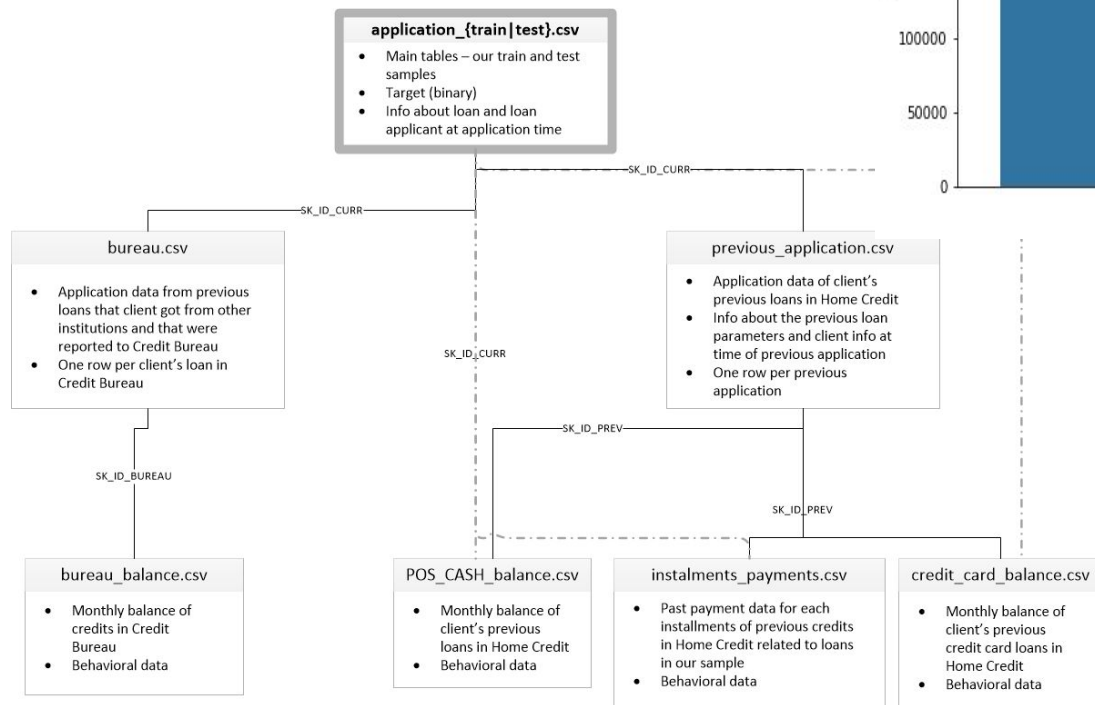
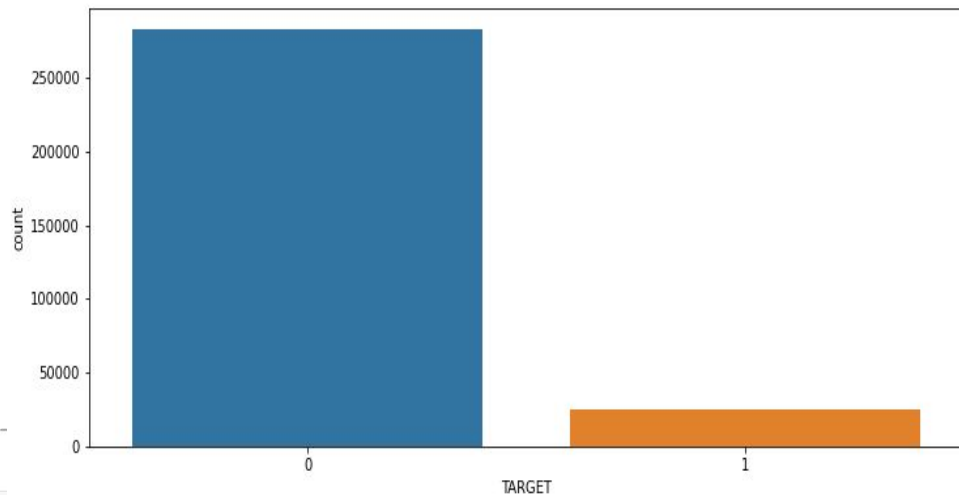
L'entreprise "**Prêt à dépenser**", qui propose des crédits à la consommation pour des personnes ayant peu ou pas du tout d'historique de prêt, souhaite mettre en œuvre un outil de “scoring crédit” pour calculer la probabilité qu’un client rembourse son crédit, puis classifie la demande en crédit accordé ou refusé. Pour ce but, l’entreprise souhaite développer un algorithme de classification en s’appuyant sur des sources de données variées (données comportementales, données provenant d'autres institutions financières, etc.).

Ce travail a pour objectif principal:

- A. Construire un modèle de scoring qui donnera une prédiction sur la probabilité de faillite d'un client de façon automatique.
- B. Construire un dashboard interactif à destination des gestionnaires de la relation client permettant d'interpréter les prédictions faites par le modèle, et d'améliorer la connaissance client des chargés de relation client.

Données

- Non équilibrées
- 307511 clients



Méthodologie d'entraînement du modèle

Test de différents schémas de préparation des données

Pour surmonter le problème des données non équilibrées, nous avons effectué la modélisation sur 3 ensembles de données clients:

- A. Non-équilibrées
- B. Équilibrées à l'aide de la méthode de sous-échantillonnage NearMiss
- C. Équilibrées à l'aide de la méthode de sur-échantillonnage SMOTE

Trois jeux de données: les jeux d'apprentissage (70%), de validation (15%) et de test (15%)

Différents algorithmes d'apprentissage

3 modèles testé :

- A. *LGBMClassifier*
- B. *CatBoostClassifier*
- C. *XGBClassifier*

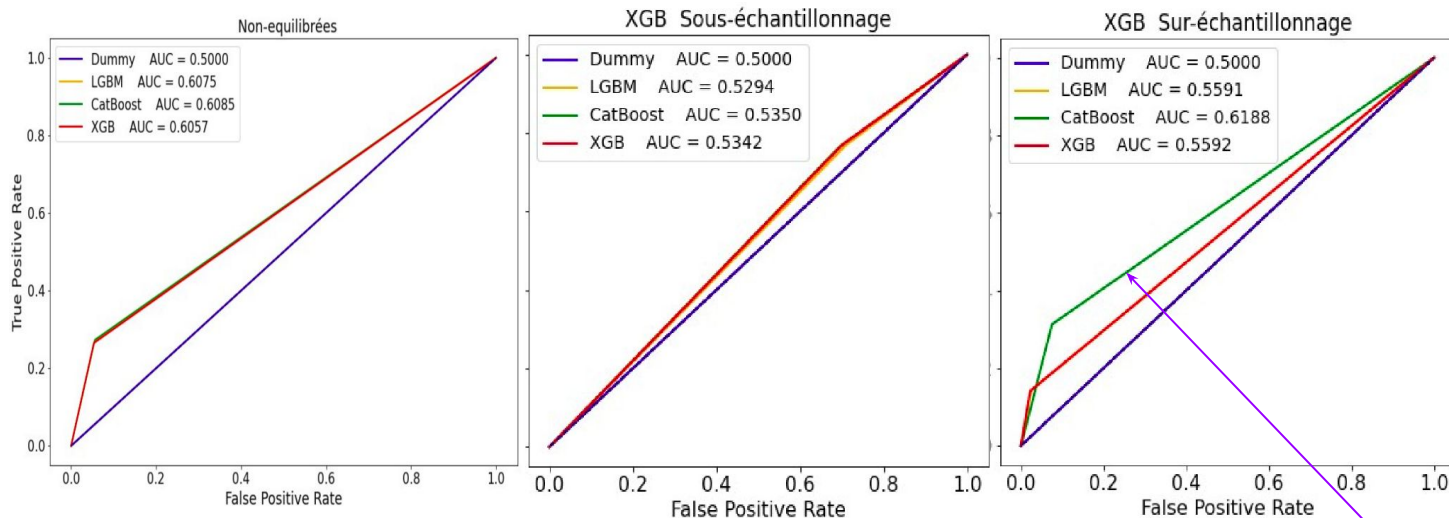
Fonction coût métier

$$F_c = -2 * (tn + tp) + fp + 10 * fn$$

tn est vrai négatif, *tp* est vrai positive,
fp est faux positive et *fn* est faux négatif.

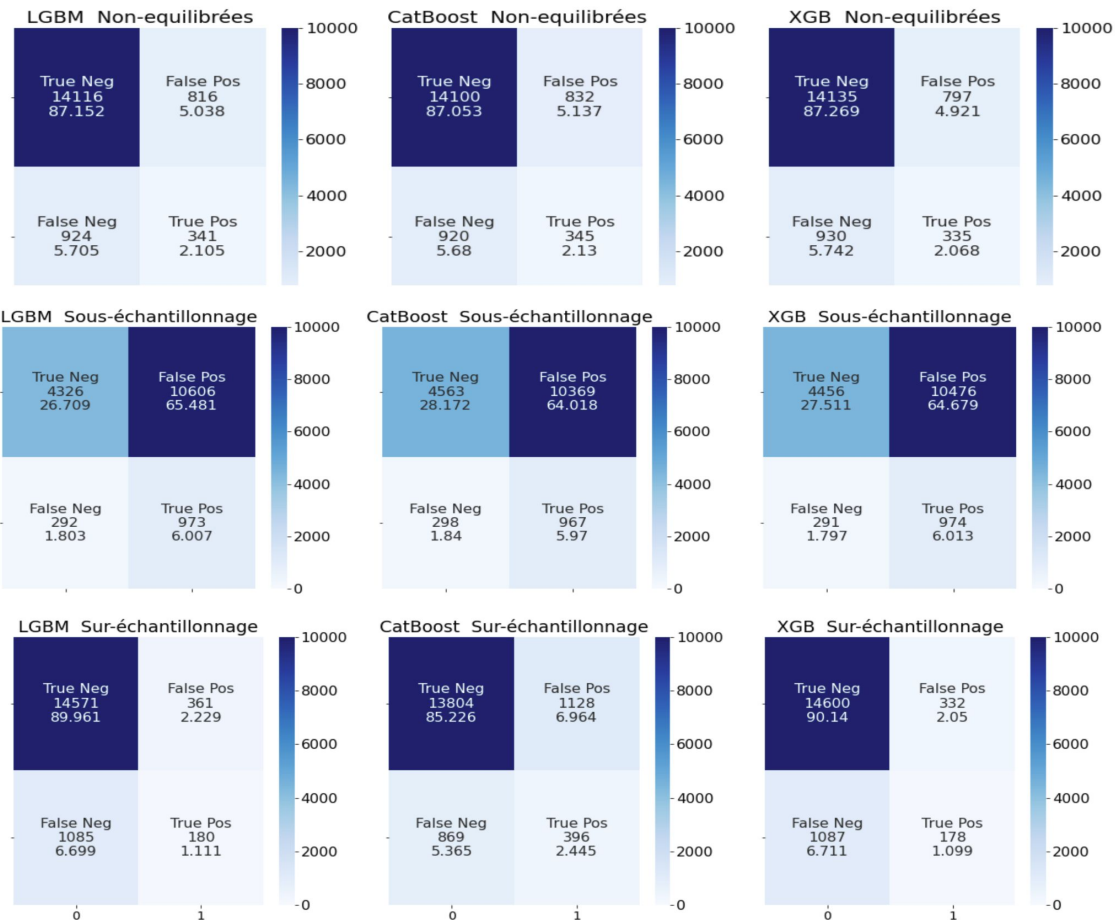
Algorithme d'optimisation

La meilleure combinaison d'hyperparamètres de chaque algorithme a été identifiée et utilisée.



Le meilleur modèle

Matrix de confusion



Présentation API et Dashboard

1. API développé avec Flask

<https://p7modelescoringapi.herokuapp.com/>

2. Dashboard développé avec Dash et Flask

<https://p7modelescoring.herokuapp.com/>

Pour visualiser l'API / le dashboard, veuillez introduire un des numéro clients fourni en haut de la barre de recherche