Note méthodologique du Projet 7 “Implémenter un modèle de scoring”

Par Malik Houni

1)Context

Ce document est une note méthodologique du projet7 “ Implémenter un modèle de scoring” du parcours Data scientist d’Openclassroom. Il présente le processus de modélisation et les outils nécessaires à la compréhension du modèle réalisé dans le cadre de ce projet.

L’entreprise “PretADépenser” 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. Elle souhaite donc développer un **algorithme de classification** en s’appuyant sur des sources de données variées (données comportementales, données provenant d'autres institutions financières, etc.).

De plus, les chargés de relation client ont fait remonter le fait que les clients sont de plus en plus demandeurs de **transparence** vis-à-vis des décisions d’octroi de crédit. Cette demande de transparence des clients va tout à fait dans le sens des valeurs que l’entreprise veut incarner.

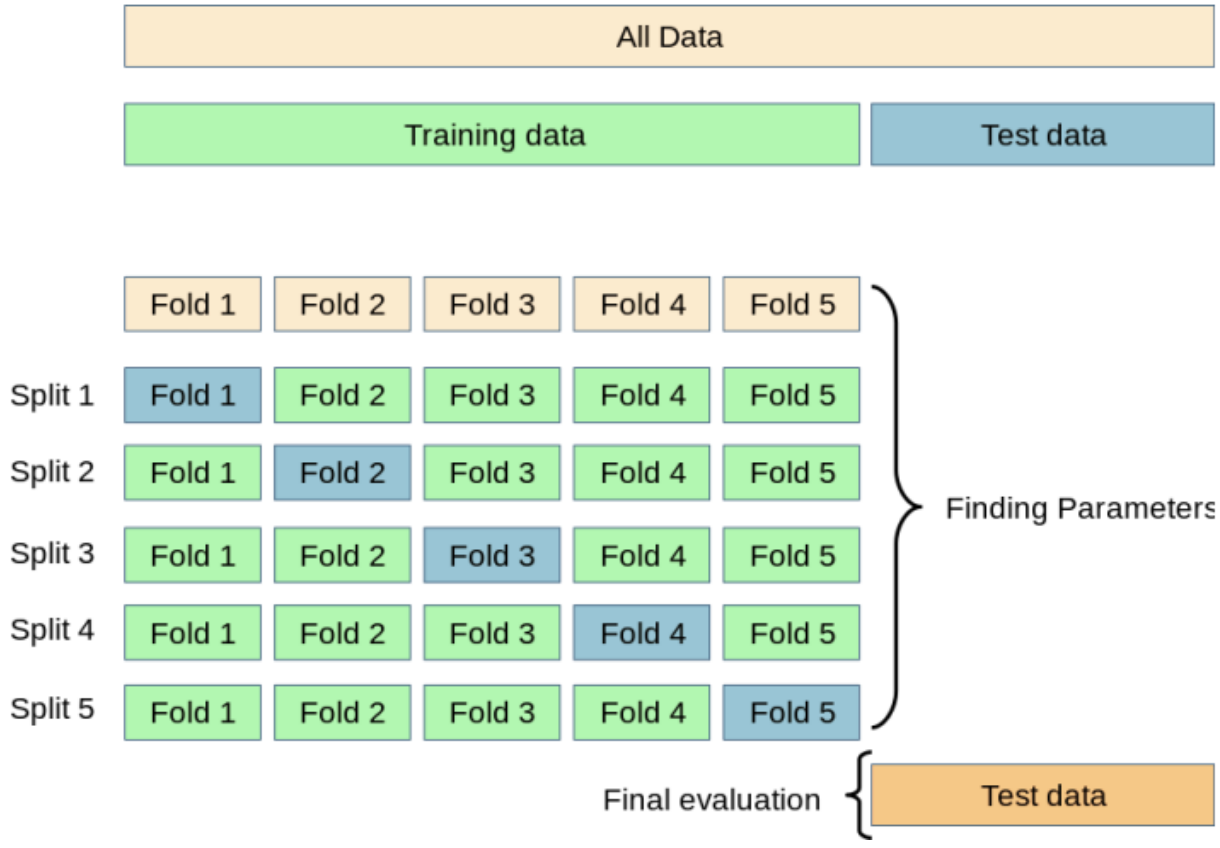
Prêt à dépenser décide donc de **développer un dashboard interactif** pour que les chargés de relation client puissent à la fois expliquer de façon la plus transparente possible les décisions d’octroi de crédit, mais également permettre à leurs clients de disposer de leurs informations personnelles et de les explorer facilement.

2) Méthodologie d'entraînement du modèle

Le modèle entraîné dans le cadre de ce projet a été entraîné sur la base du jeu de données après analyse exploratoire et création de nouvelles features engineered. Plusieurs notebooks sont présents sur kaggle pouvant servir de colonne vertébrale de ce projet.

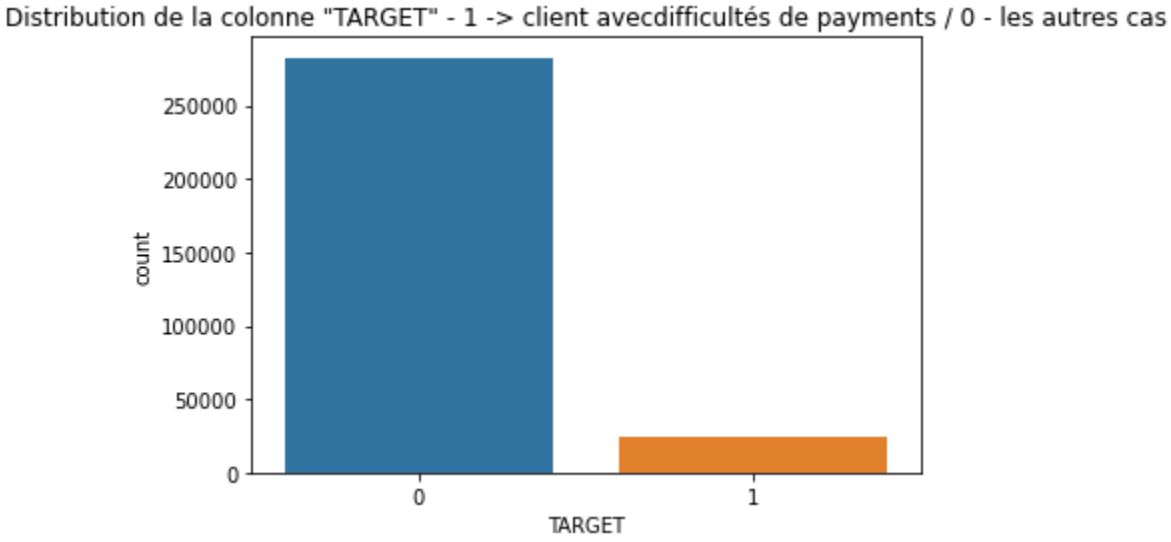
Le jeu de données initial a été séparé en plusieurs parties de façon à obtenir

- un jeu de données prévu pour l'entraînement (80% des individus) qui a été séparé en plusieurs folds pour entraîner les différents modèles choisis et optimiser les paramètres (cross validation) sans overfitting.
- un jeu de test (20 % des individus) pour l'évaluation finale du modèle.



Le problème est un problème de classification binaire(résultat sous forme de 1 ou 0 dans la colonne Target).Nous avons un ensemble de personnes en forte minorité (9

% de clients en défaut contre ~91 % de clients sans défaut).



Nous avons donc un déséquilibre de classe qui doit être pris en compte dans l'entraînement des modèles . En effet nous serions dans la possibilité de prédire systématiquement que les clients sans défaut aurait une précision de 92 % et pourrait être considéré faussement comme un modèle performant alors qu'il n'engendrent pas la détection de clients à risque.

Différents modèles ont été testés avec recherche d'hyper paramètres et cross validation (5 folds) :

- Dummy Classifier (baseline)
- Random Forest Classifier
- LightGBM
- XGBoost

3) Algorithme d'optimisation et métrique d'évaluation

Les modèles ont été entraînés dans la fonction de cross validation et testés suivant différentes combinaisons d'hyper paramètres.
Les fonctions de coût pour les algorithmes entraînés sont les suivantes :

Algorithme	Fonction de coût
Random Forest	Minimisation du coef de GINI
LightGBM	Nombre de noeuds
XGBOOST	Gradiant Stochastique

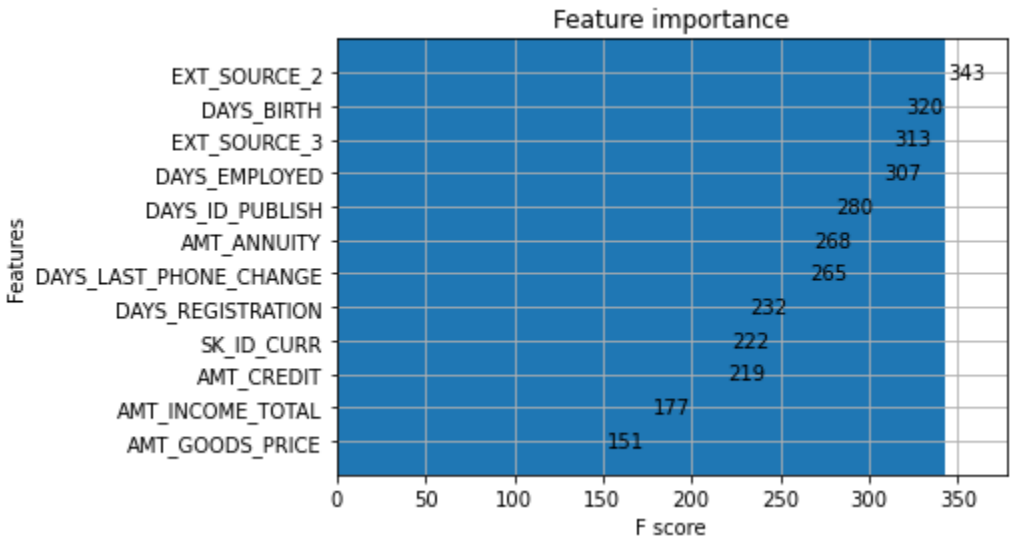
Choix de la métrique:
Nous utiliserons le ROC -AUC ainsi que le score F1 comme métrics.
Dans le jeu de données de base, il y a une part de 92 % des clients qui n'ont pas d'incident de paiement, tandis que 8 % des clients ont eu des incidents.
La matrice de confusion est la suivante :

	Clients prédit en défaut	Clients prédits sans défauts
Clients vraiment en défaut	Vrai +	Faux -
Clients sans défaut	Faux +	Vrai -

On cherchera donc à minimiser le pourcentage de faux négatifs et à maximiser le pourcentage de vrais positifs.
De plus nous donnerons des résultats sous forme de pourcentage et non de valeurs binaire (0 ou 1)

4) Interprétabilité du modèle

Le modèle étant destiné à des équipes opérationnelles devant être en mesure d'expliquer les décisions de l'algorithme à des clients réels, le modèle est accompagné de détails d'observations sur les différents aspects des datasets. Les opérations de transformations sont détaillées dans le notebooks ainsi que l'impact sur le reste du travail par rapport au modèle choisi. En utilisant les metrics retenus nous obtenons des résultats pouvant être interprété assez clairement.
L'utilisation des features importance permettra de mettre l'accent sur les features les plus à même à influencer le calcul de prédiction de repayment.



Par la suite les résultats des différents modèles aiguillent sur le choix:

Les résultats sont les suivants:

modèle	Score ROC AUC	F1 Score
Dummy classifier	0.5	0.88
Random Forest	0.5	0.88
LightGBM	0.68	0.76
XGBOOST	0.67	0.80

5) Limites et améliorations possibles

La modélisation pourrait être augmentée par l'utilisation des autres datasets. D'autres algorithmes peuvent venir ajouter une couche de recherche sur l'optimisation des résultats et sur la prédiction des résultats. Cela peut avoir un impact quant au déploiement d'une API et Dashboard dédié. Le choix de l'algorithme de machine learning peut impacter le temps de travail au niveau de l'API hébergé dont le travail de prédiction est dit "live". Le dashboard n'est pas impacté mais peut être amélioré en utilisant plus l'API de simulation.

Sources:

- https://blog.ekinox.io/ml/gradient-boosting
- https://scikit-learn.org/stable/modules/generated/sklearn.ensemble.RandomForestClassifier.html
- https://towardsdatascience.com/a-machine-learning-approach-to-credit-risk-assessment-ba8eda1cd11f
- https://openclassrooms.com/fr/paths/164/projects/632/assignment