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

Par Malik Houni

1)Context

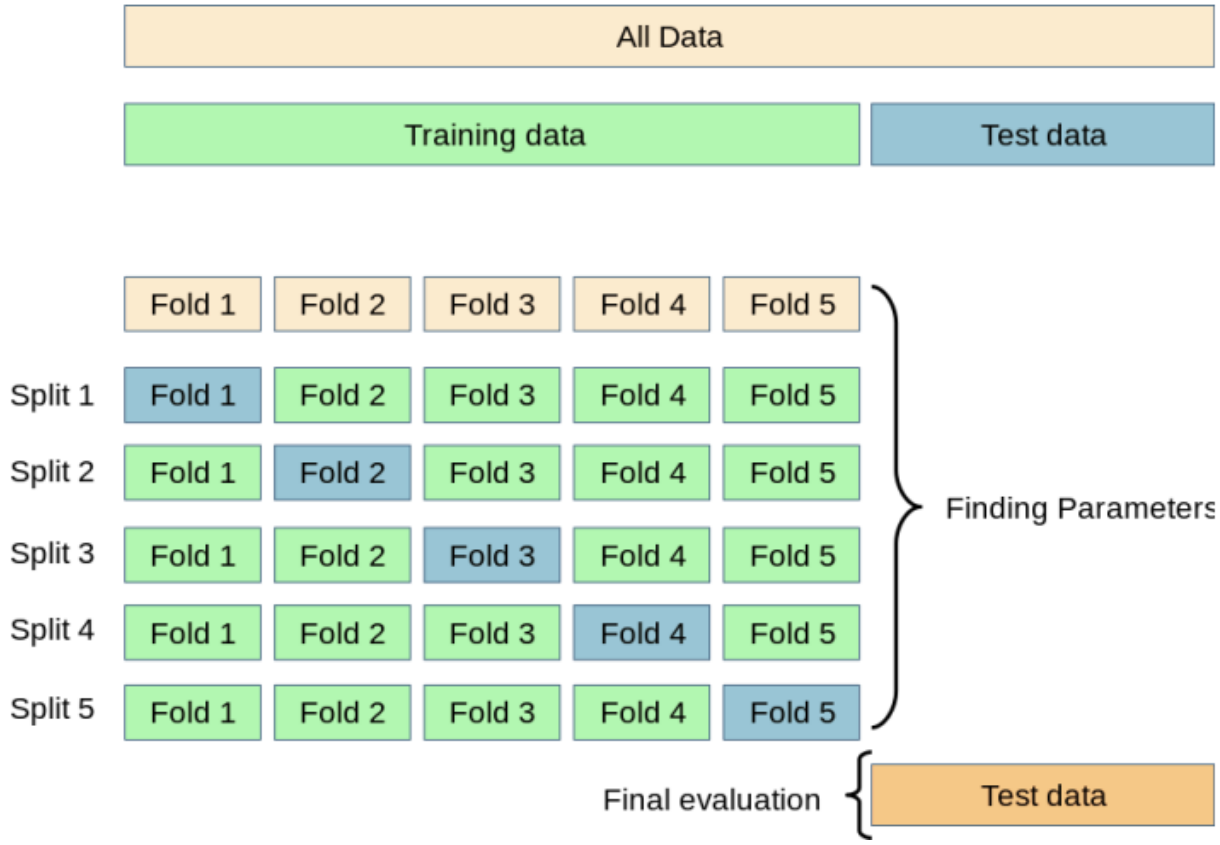
Ce document est un livrable du projet7 “ Implémenter un modèle de scoring” du parcours Data scientist d’Openclassroom. Il présente le processus de modélisation et d’intreprétabilité du modèle réalisé dans le cadre de ce projet.
L’entreprise souhaite **mettre en œuvre un outil de “scoring crédit” pour 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. Le notebook utilisé est consultable sur le site Kaggle. Le jeu de données initial a été séparé en plusieurs parties de façon à disposer : D'un jeu de training (75% des individus) qui a été séparé en plusieurs folds pour entraîner les différents modèles et optimiser les paramètres (cross validation) sans overfitting. D'un jeu de test (25 % des individus) pour l'évaluation finale du modèle



Le problème est un problème de classification binaire avec une classe sous représentée (9 % de clients en défaut contre 91 % de clients sans défaut). Ce déséquilibre des classes doit être pris en
compte dans l'entraînement des modèles puisqu'un modèle « naif » prédisant systématiquement que les clients sont sans défaut aurait une accuracy (justesse) de 0.92 et pourrait être considéré à tort comme un modèle performant alors qu'il ne permettrait pas de détecter les clients à risque.

Deux approches pour rééquilibrer les deux classes ont été testées : sampling based, et cost function based

Pour chaque approche, différents modèles ont été testés avec recherche d'hyperparamètres et cross validation (5 folds) :

- Random Forest Classifier
- LightGBM
- XGBoost

3) Fonction coût, algorithme d'optimisation et métrique d'évaluation

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

Algorithme	Fonction de coût
Random Forest	Minimisation du coef de GINI
LighGBM	Nombre de noeuds où la feature est utilisée
XGBOOST	Regr Logistique

Choix de la métrique

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.

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 pour expliquer.

Pour réaliser ce module, la première perspective envisagée est d'utiliser l'importance des features issues des différents modèles utilisés mais cette approche n'est pas optimale car Les features importances en sortie de modèle sont difficiles à interpréter lorsqu'il y a des variables issues de One Hot Encoding

5) Limites et améliorations possibles

La modélisation pourrait être augmentée par l'utilisation des autres datasets. D'autre algorithmes peuvent venir ajouté une couche de recherche sur l'optimisation des résultats sur la prédiction des résultats

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