**Note méthodologique**

* **Partie 1 : Méthodologie d’entraînement du modèle**

Nous entraînons un modèle de classification binaire pour établir si un client est à risque de défaut de paiement pour un crédit (valeur 1) ou non (valeur 0). Nous allons comparer plusieurs modèles : DummyClassifier, LogisticRegression, LightGradientBoostingModel, et RandomForestClassifier.

Pour sélectionner le meilleur modèle, nous utilisons une méthode de validation croisée avec RandomizedSearch. Cela nous permet d'explorer différentes combinaisons d'hyperparamètres pour chaque modèle et d'évaluer leur performance.

Après avoir sélectionné le meilleur modèle, nous identifions les dix features les plus importantes. Ensuite, nous réentraînons le modèle en utilisant uniquement ces dix features et calculons le seuil de probabilité de prédiction pour lequel la fonction de coût est minimisée.

* **Partie 2 : Traitement du déséquilibre des classes**

Dans cette partie, nous abordons le déséquilibre des classes dans nos données, avec seulement 7-8% de clients à risque de défaut de paiement dans la base. Pour rétablir l'équilibre, nous appliquons la méthode SMOTE (Synthetic Minority Over-sampling Technique) avec un SVM (Support Vector Machine).

Le SMOTE permet de générer des échantillons synthétiques pour les clients à risque de défaut de paiement afin d'augmenter leur nombre et d'équilibrer les classes. En utilisant un SVM, nous apprenons à séparer efficacement les deux classes équilibrées pour la classification.

* **Partie 3 : fonction de coût métier, algorithme d’optimisation et métrique d’évaluation**

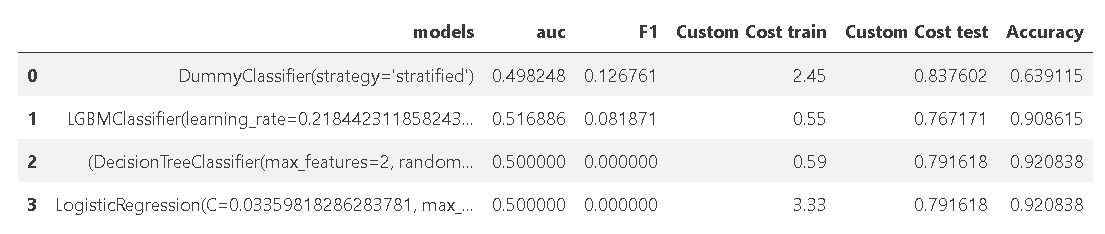
Une mauvaise catégorisation des clients entraîne des coûts à l’organisme de crédit dans une proportion de un client à risque mal identifié (Taux de Faux Négatifs) pour dix clients sûrs considérés à tort comme risqués (Taux de Faux Positifs). Ce qui nous donne la fonction de coût suivante :

*Loss = 10\*FPR+FNR*

L’algorithme d’optimisation retenu à l’issue de l’entraînement et du test des données est le Light Gradient Boosting Model. C’est un modèle de boosting itératif avec arbres de décision peu profonds, optimisé pour accélérer l'apprentissage et améliorer la précision en se concentrant sur les échantillons difficiles.

La métrique d’évaluation employé est la minimisation de la fonction de coût.

* **Partie 4 : Tableau de synthèse des résultats**

****

* **Partie 5 : Interprétabilité globale et locale du modèle**

Tout d'abord, nous comparons les clients sur les dix features les plus importantes par rapport à la population globale. Cela nous permet de déterminer de manière globale les caractéristiques clés qui distinguent les clients à risque de défaut de paiement des autres.

Ensuite, nous analysons les clients par rapport à des sous-populations spécifiques, en les comparant à des clients du même sexe et de la même tranche d'âge, à plus ou moins cinq ans près. Cela nous donne une perspective locale et nous aide à comprendre comment les caractéristiques importantes peuvent varier selon ces sous-groupes.

* **Partie 6 : Limites et améliorations possibles**

Parmi les limites potentielles, nous pouvons mentionner la sensibilité des modèles aux choix des hyperparamètres et le fait que la performance du modèle pourrait varier dans le temps en raison de l'évolution des données.

Quant aux améliorations possibles, nous pourrions explorer d'autres techniques de traitement.

* **Partie 7 : Analyse du Data Drift**

L'analyse de la dérive des données sur les features importantes révèle des résultats significatifs. Quatre colonnes présentent une dérive très prononcée avec un p-value de K-S à 0, indiquant des différences significatives entre les ensembles de données de référence et actuels. Ces différences nécessitent une attention particulière car elles peuvent affecter les performances du modèle. De plus, quatre autres colonnes montrent une dérive modérée avec des p-values de K-S allant de 0,01 à 0,24. Bien que moins marquées, ces variations doivent également être surveillées. Cependant, les deux colonnes avec des p-values de Z-test à 0,6 et 0,96 ne montrent pas de signes de dérive significative. Dans l'ensemble, ces résultats soulignent l'importance de surveiller et d'ajuster les colonnes présentant une dérive marquée pour maintenir la performance du modèle.