Note Méthodologique : Modèle LightGBM pour le Classement des Clients d'une Banque

1. Méthodologie d'Entraînement du Modèle

1.1. Collecte et Préparation des Données

Les données utilisées proviennent des sources de données variées (données comportementales, données provenant d'autres institutions financières, etc.), comprenant des variables telles que le revenu, le montant des crédits, l'historique de remboursement, et des données démographiques.

Les étapes de préparation incluent :

- **Encodage des Variables Catégorielles** : Utilisation de l'encodage one-hot pour les variables catégorielles.
- **Consolidation des Données** : Unifier les sept (7) data sets initiaux en une seule dataframe.
- **Séparation des variables** : Séparer les features de la variable target
- **Imputation de données** : Utilisation de SimpleImputer pour Remplacer les valeurs manquantes par la moyenne de la variable
- **Normalisation** : Standardisation des variables numériques pour garantir une échelle similaire.

1.2. Division des Données

Les données sont divisées en trois ensembles :

- **Ensemble d'Entraînement (**Pour ajuster le modèle**)** : Les lignes de la dataframe des features dont la valeur de la colonnes Target ne sont pas manquantes
- **Ensemble de Validation ()** : Pour optimiser les hyperparamètres.
- **Ensemble de Test (**Pour évaluer les performances finales du modèle**)** : Les lignes de la dataframe des features dont la valeur de la colonnes Target sont manquantes
- **1.3. Entraînement du Modèle**

LightGBM, abréviation de **Light Gradient Boosting Machine,** est basé sur l'algorithme de Gradient Boosting et utilise des arbres de décision. LightGBM est choisi pour son efficacité et sa rapidité avec de grandes quantités de données. Les hyperparamètres sont optimisés via la recherche en grille, en se concentrant sur :

- **Learning Rate** : Pour contrôler la vitesse d'apprentissage.

- **Nombre d'Estimates** : Pour éviter le surajustement.
- **Max Depth** : Pour contrôler la complexité du modèle.

1.4. Validation Croisée

Une validation croisée en 5-plis est mise en place pour évaluer la robustesse du modèle et éviter le surajustement. Les performances sont mesurées sur chaque pli et moyennées.

2. Traitement du Déséquilibre des Classes

Le dataset présente un déséquilibre entre les classes (clients bons vs clients mauvais). Les clients bons sont largement minoritaires aux clients bons.

Pour traiter ce déséquilibre :

- **Sur-échantillonnage** : Application de la technique SMOTE (Synthetic Minority Over-sampling Technique) pour générer des exemples synthétiques de la classe minoritaire afin de résoudre le problème de déséquilibre.

Cette technique permet d'améliorer la capacité du modèle à prédire correctement les clients à risque.

3. Fonction Coût Métier, Algorithme d'Optimisation et Métrique d'Évaluation

3.1. Fonction Coût Métier

La fonction coût métier est basée sur le coût des faux positifs et faux négatifs. Un faux positif (FP) représente un client classé comme bon alors qu'il ne l'est pas, entraînant une perte financière. Un faux négatif (FN) représente un client classé comme mauvais alors qu'il est bon, ce qui peut entraîner une perte d’opportunités.

Nous avons considéré le coût d’un FN est dix fois supérieur au coût d’un FP

3.2. Algorithme d'Optimisation

LightGBM utilise un algorithme de gradient boosting par arbres de décision, qui optimise la fonction de coût en mettant à jour les poids des arbres successifs pour minimiser l'erreur.

3.3. Métrique d'Évaluation

Les métriques clés incluent :

- **AUC** : Pour évaluer la capacité du modèle à distinguer entre les classes.
- **Accuracy** : le rapport entre les prédictions correctes et le nombre total de prédictions

4. Tableau de Synthèse des Résultats

Métrique	Modèle avec deséquilibre	Modèle équilibré par SMOTE	Modèle avec score métier
AUC	0,731	0,889	0,731
Acuuracy	0,918	0,953	0,920

5. Interprétabilité Globale et Locale du Modèle

5.1. Interprétabilité Globale

L'importance des caractéristiques est analysée via l'importance des variables, permettant de visualiser quelles variables influencent le plus le modèle.

5.2. Interprétabilité Locale

Des outils comme SHAP (SHapley Additive exPlanations) sont utilisés pour expliquer les prédictions individuelles, offrant des insights sur l'impact de chaque caractéristique sur la décision du modèle.

6. Limites et Améliorations Possibles

6.1. Limites

- **Sensibilité aux Hyperparamètres** : La performance dépend fortement de l'optimisation des hyperparamètres.

6.2. Améliorations

- **Mise à Jour Continue** : Réentraîner régulièrement le modèle avec de nouvelles données pour s'adapter aux changements du marché.

7. Analyse du Data Drift

Une surveillance continue est mise en place pour détecter le data drift. Cela inclut :

- **Statistiques descriptives :** Comparer les statistiques descriptives (moyenne, écart-type, etc.) des nouvelles données avec celles des données d'entraînement.
- **Métriques de Performance** : Suivi des métriques de performance du modèle au fil du temps pour identifier toute dégradation.
- **Sélection de caractéristiques :** Identifier les caractéristiques les plus stables et les moins sensibles au drift.

Une action corrective, comme le réentraînement du modèle, sera initiée si un drift significatif est détecté.

Le drift des données est un défi majeur en machine learning. Pour garantir la fiabilité des modèles, il est essentiel de mettre en place des mécanismes de détection et de gestion du drift. En combinant des techniques de monitoring, de retraitement des données et d'apprentissage continu, il est possible de construire des modèles robustes et adaptatifs.