Projet Master IREF : Prédiction du risque de crédit

Prérequis :

- Groupe de 3 ou 4 étudiants
- Langage de programmation Python
- Utilisation des notebooks avec Jupyter
- Librairies recommandées :
    - Manipulation de données : pandas / numpy
    - Régression logistique / Tests statistiques : statsmodels / scipy
    - Machine Learning : scikit-learn
    - Visualisation : matplotlib / seaborn / plotly

Le projet devra être rendu via des notebooks, pensez donc bien à nettoyer, commenter et structurer vos notebooks. Un document PDF est attendu pour que vous présentiez vos résultats (maximum 20 pages).

Date du rendu du projet : 03/02/2025 (les projets en retard ne seront pas notés)

Données :

Dans ce projet, vous travaillez pour une Banque vous demandant d'élaborer un outil automatisant le process d'octroi de crédit à ses futurs clients. La banque dispose d'un historique des crédits effectués par ses clients, avec des informations personnelles sur le client et s'il a fait défaut ou non pendant la période de remboursement du crédit. Les données contiennent les informations ci-dessous :

- SK_ID_CURR : identifiant du client
- GOOD_PAYER : 1 s'il n'y a pas eu de retard de remboursement, 0 sinon
- CODE_GENDER : le genre du client
- FLAG_OWN_CAR : le client possède une voiture
- FLAG_OWN_REALTY : le client est propriétaire
- CNT_CHILDREN : nombre d'enfant du client
- AMT_INCOME_TOTAL : les revenus du client
- AMT_CREDIT : le montant du crédit
- AMT_GOODS_PRICE : le montant du produit pour lequel le crédit a été pris
- NAME_INCOME_TYPE : le type de revenus du client
- NAME_EDUCATION_TYPE : niveau académique du client
- NAME_FAMILY_STATUS: status familial du client
- NAME_CONTRACT_TYPE: crédit comptant ou en revolving
- NAME_HOUSING_TYPE: situation habitat

- TOTALAREA_MODE: surface normalisée d'habitation
- DAYS_BIRTH: Age du client
- DAYS_EMPLOYED : nombre d'année consécutif du dernier emploi du client
- OCCUPATION_TYPE : profession du client
- ORGANIZATION_TYPE : secteur d'emploi
- EXT_SOURCE_1 : score de crédit bureau 1
- EXT_SOURCE_2 : score de crédit bureau 2
- EXT_SOURCE_3 : score de crédit bureau 3
- AMT_REQ_CREDIT_BUREAU_YEARS: nombre de demande de crédit effectué par le client dans l'année précédente

La Banque vous fixe les objectifs suivants :

1) **Effectuer une analyse descriptive des données bancaires**
2) **Réaliser une grille de score pour l'octroi des crédits à la consommation**
3) **Proposer un modèle de Machine Learning pour challenger la grille de score**

La notation du projet prendra en compte les 3 objectifs définis ainsi que la qualité du code, des notebooks et du rapport.

Objectifs 1 : analyser les données dans le but de décrire le portfolio des clients pour la Banque

- Analyse univariée et bivariée :
    - Moyenne, médiane, quartile, comptage
    - Corrélation : variables explicatives et la variable cible (anova, pearson)
    - Distribution des variables (ex : kurtosis, distribution normal)
    - Analyse des valeurs extrêmes / aberrantes (ex : Interquartile Range Method)
    - Traitement des valeurs manquantes
    - Taux de risque / %BP et %MP selon les modalités (Analyse Marginale)
- Visualisation des données : boxplot, histogramme, piechart, nuage de point
- Calcul de métrique d'importances des variables :
    - WOE, IV, GINI
    - Test de significativité

Objectifs 2 : élaborer une grille de score à partir de la régression logistique

- Préparer une base de train, de validation et de test
    - 60% / 20% / 20%
    - Stratifier les clients dans l'ensemble des échantillons

    - Des techniques d'oversampling ou d'undersampling peuvent être testées (ex = SMOTE) dans le cas d'un déséquilibre du jeu de données
- Transformer les variables catégorielles et les variables numériques
    - Catégorielles : regroupement expert / WOE / taux de risque
    - Numériques : optimisation des tranches via l'indice de GINI ou regroupement par expertise
    - Vérifier la valeur d'information avant et après transformation
    - Vérifier la volumétrie des tranches créées (éviter des modalités avec moins de 5% de la population)
    - Appliquer la méthode d'indicatrice imbriquée avant l'intégration dans le modèle
- Entraîner un modèle de régression logistique
    - Méthode ascendante
    - Ajout des variables ayant un dIV > 0,02 et tester la significativité des distributions réelles et estimées
    - Liberté du choix des variables selon vos analyses
    - Vérifié la significativité des coefficients → supprimer ou ajouter en conséquence de nouvelles variables
- Grille de score :
    - Maximum de 6 variables à intégrer
    - Normaliser la grille :
        - Valeur de référence = 500
        - Facteur multiplicatif = 2
        - Nombre de points = 20
        - Ratio de référence = 1
    - Vérifier la monotonie des points attribuées aux modalités des variables et les taux de risques

- Performance du modèle :
    - Calculer le GINI du jeu d'entrainement, de validation et du jeu de test
    - Afficher la courbe de concentration associée
    - Créer une table des déciles des tranches de scores

    - Déterminer la tranche maximisant le KS = $\max|\%BAD_{cum} - \%GOOD_{cum}|$
    - Etudier d'autres métriques dans vos analyses : R2 ajustés, AIC et BIC
- Calculer l'impact économique de la grille de score et déterminer le seuil de décision :
    - Un client faux positif = pertes de la totalité du crédit financé
    - Un client faux négatif = pertes de 8% (taux d'intérêt des crédits)
    - Choisir la note de score maximisant le gain ou minimisant la perte pour déterminer le seuil de décision pour refuser ou accepter un crédit
    - Tracer une courbe de l'équilibre financier en fonction du seuil

Objectifs 3 : challenger la grille de score avec un modèle de Machine Learning

- Recommandation :
    - Modèle simple : Naives Bayes / Arbre de décision / K-nearest Neighbor
    - Modèle complexe : **Random Forest / Gradient Boosting** / Réseau de neurones
- **Plusieurs algorithmes peuvent être testés**
- Transformation des variables :
    - Catégorielles : OneHotEncoding, BinaryEncoding, TargetEncoding, WOE Encoding
    - Numériques : utilisation des variables brutes
- Optimisation des hyperparamètres :
    - Utiliser la cross-validation sur le jeu de train
    - Gridsearch, RandomSearch ou recherche optimisée : exemple de package hyperopt ou optuna
    - Choisir la métrique d'optimisation : précision, rappel, F1_score, fonction de coût personnalisée
- Performance du modèle :
    - Calculer le GINI du jeu d'entrainement, la moyenne des folds sur la cross-validation et du jeu de test
- Impact économique :
    - Effectuer le même calcul d'impact économique en déterminant le seuil adéquat de refus ou d'acceptation du crédit
- Interprétabilité du modèle de Machine Learning

- Utiliser une technique de feature importance pour expliquer les variables ayant le plus d'impact dans la décision d'octroi
- Les modèles à base d'arbre ont une méthode interne de feature importance basé sur la réduction d'impureté
- Des techniques agnostiques aux modèles existent : Shapley Values (recommandé) ou Permutation Feature Importance
- Il est possible d'optimiser la performance de son modèle en le réentraînant à partir des meilleures variables calculées