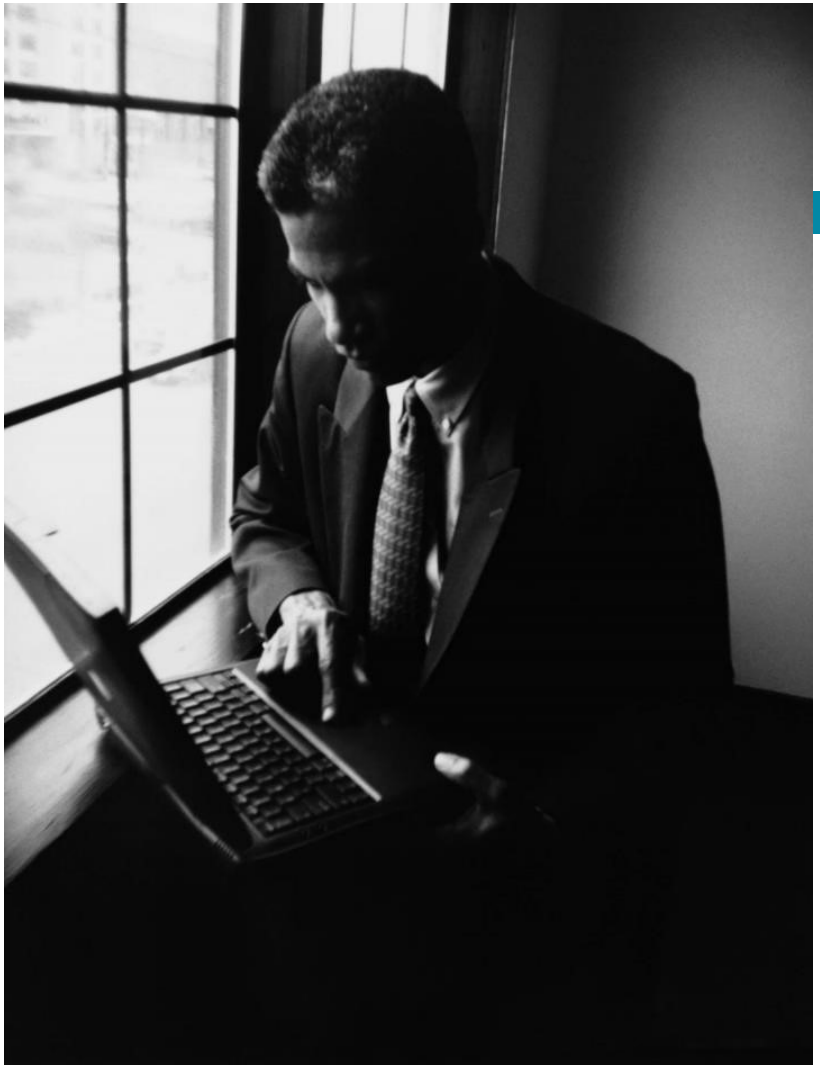
IMPLEMENTER UN MODELE DE SCORING

Corentin Jay – OpenClassrooms Data Science – Projet 7



Détails du projet

ENJEU

Mettre en place un outil de scoring crédit pour calculer la probabilité qu'un client rembourse son crédit à la consommation, puis classifie la demande en crédit accordé ou refusé.

DÉPLOIEMENT

Développer un algorithme de classification, et mettre en place un dashboard intéractif à l'attention des chargés de relation client, pouvant être présenté au client.

SPECIFICATIONS

- Permettre de visualiser le score et son interprétation ;
- Permettre de visualiser les informations descriptives d'un client ;
- Permettre de comparer les informations d'un client à l'ensemble des clients ou à un groupe de clients similaires.

DATASET

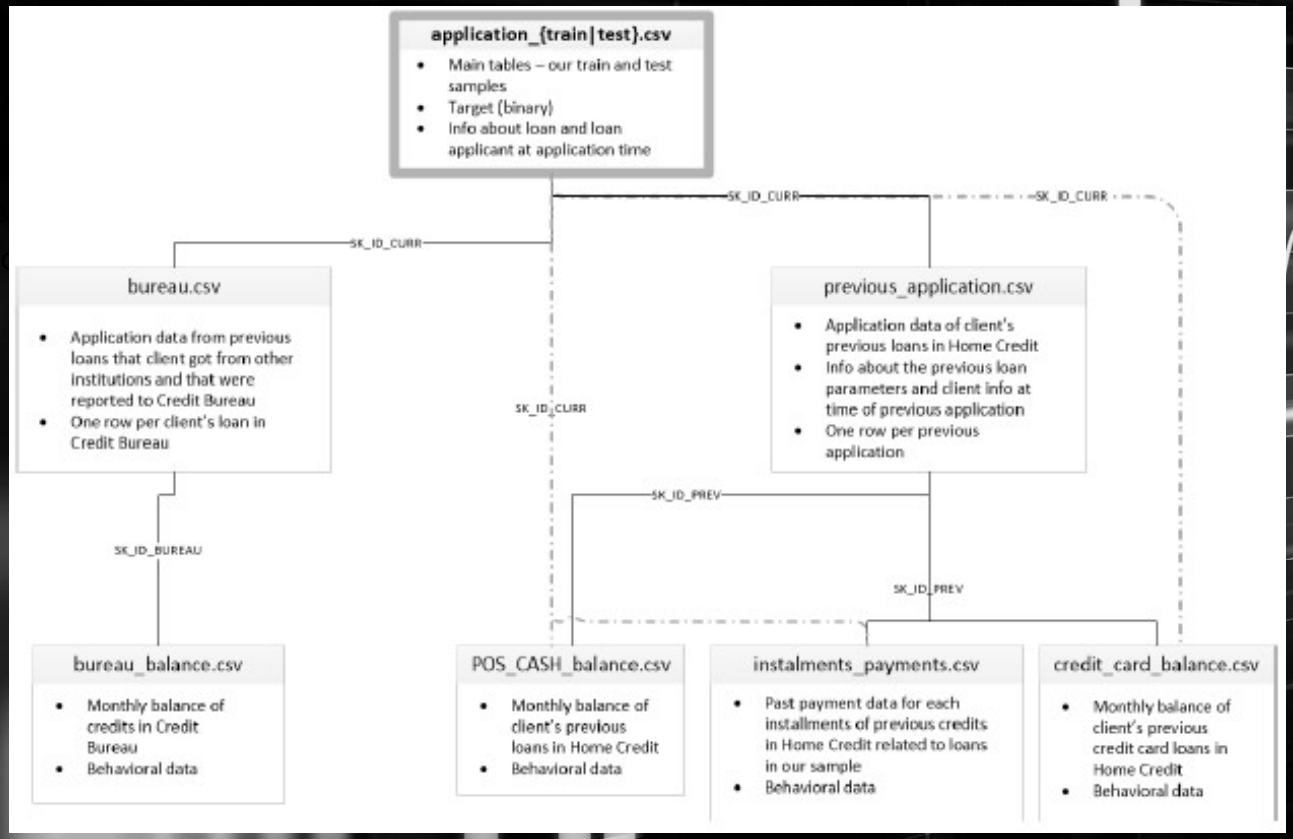
Dataset initial composé de 10 fichiers (.csv) :

- application_test
- application_train
- bureau
- bureau_balance
- credit_card_balance
- Homecredit_columns_description
- installments_payments
- POS_CASH_balance
- previous_application
- sample_submission

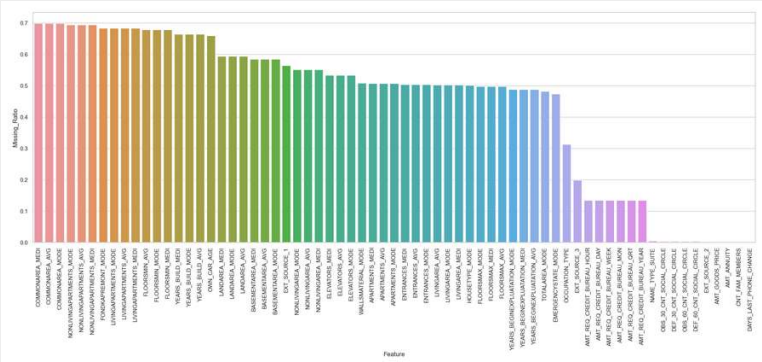
NB :

- Le Dataset est déjà splité en échantillons d'entraînement et de test
- Clés communes : SK_ID_CURR (ID client), SK_ID_BUREAU (agence) et SK_ID_PREV
- Echantillon d'entraînement composé de 307'511 individus, et échantillons de test composé de 48'744 individus
- Features numériques et catégorielles (total 121)
- Target : crédit en défaut (1) ou non (0)
- Poids total : 2,49Go

ARCHITECTURE DU DATASET

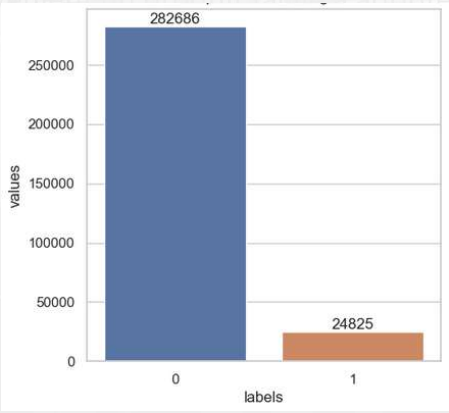


DATA VISUALIZATION



Dans le fichier application_train, 67 features contiennent au moins une valeur nulle.

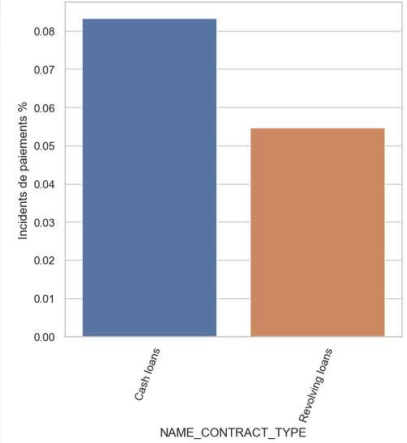
On observe beaucoup plus de crédits sans défaillance (target = 0), à hauteur de 91,93%, la target n'est pas équilibrée.



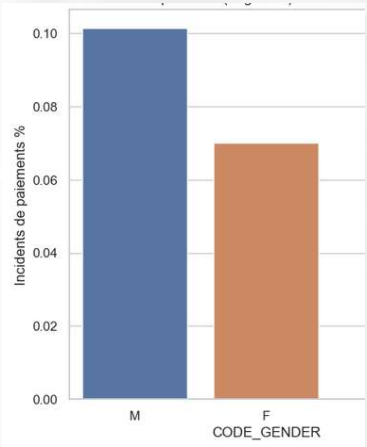
DATA VISUALIZATION

DISTRIBUTION DES DEFAILLANCES SELON LES DIFFERENTES FEATURES (EXEMPLES) :

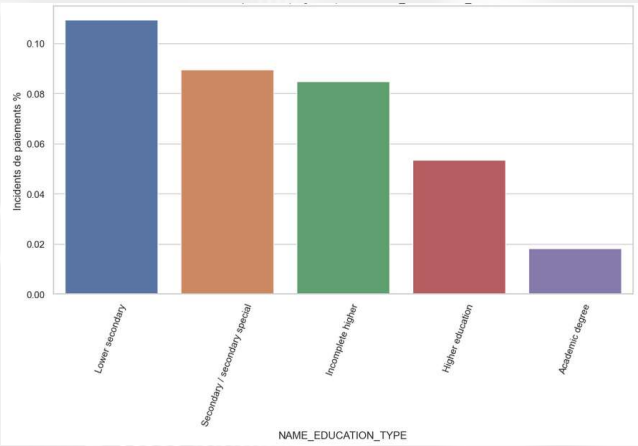
Type de prêt



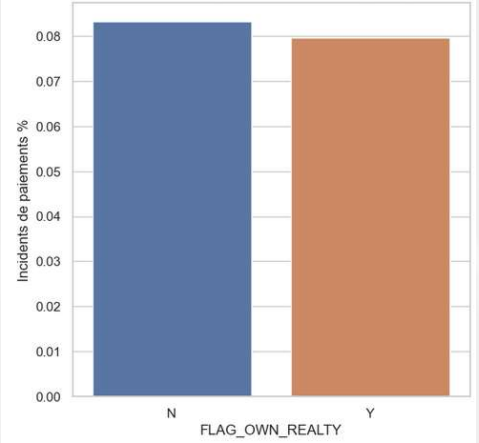
Sexe



Niveau d'étude



Propriétaire



FEATURE ENGINEERING

- Retrait des valeurs aberrantes (ex : CODE_GENDER = XNA) et des valeurs infines
- Traitement des outliers (ex : DAYS_EMPLOYED = 365’243)
- Création de features (ex : DAYS_EMPLOYED_PERCENT, ANNUITY_INCOME_PERCENT, INCOME_CREDIT_PERCENT)
- Fusion des différents fichiers selon les clés communes
- Aggrégation des valeurs (Min, Max, Mean, Sum, Var)
- Encodage des features catégorielles (OneHotEncoder) et imputation valeurs nulles (médiane)
- Etude de la corrélation des features avec la Target
- Sauvegarde du dataset train après feature engineering

DEMARCHE DE MODELISATION

- Equilibrage des données :

  Limiter le nombre de clients à risque prédits comme non risqués (faux négatifs) car le risque de perte en capital est plus important pour la banque, en comparaison à la perte de chiffre d'affaires générée par les clients non risqués prédits comme risqués (faux positifs).

- Création d'un score client (business_score)
- Modèles testés

  RandomForestClassifier, XGBoostClassifier, LightGBM, CatBoostClassifier

- Analyse des résultats

  Comparaison des différentes métriques retenues (business_score, temps de traitement, score AUC, Recall, Precision) et sélection du meilleur modèle.

- Optimisation (GridSearchCV)

  Recherche d'optimisation des hyper-paramètres du modèle retenu

- Interprétabilité des résultats

  Analyse de l'importance des features (SHAP)

EQUILIBRAGE DES DONNEES

Solutions d'équilibrage retenue (via pipeline) :

- **SMOTE** (Synthetic Minority Oversampling Technique)
  Synthétise des nouveaux éléments pour la classe minoritaire (ici la défaillance du crédit), basés sur les valeurs des variables des k voisins les plus proches.
- **RandomUnderSampler**
  Sous pondère la classe majoritaire par sélection aléatorie d'échantillon.

	Répartition initiale	Répartition post SMOTE
Target 0 (non défaillance)	226 145	19 860
	91,93%	8,07%
Target 1 (défaillance)	45 228	22 614
	66,67%	33,33%

L'intérêt principal est d'éviter l'overfitting (ou sur-entraînement).

BUSINESS SCORE

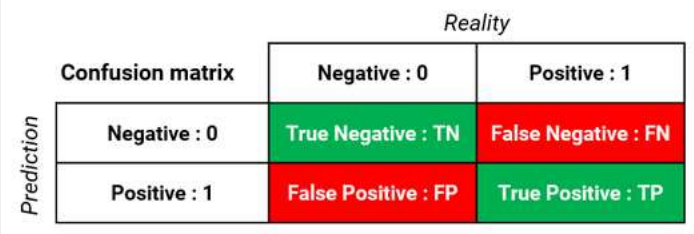
DEFINITION DE LA FONCTION RELATIVE AU BUSINESS SCORE

Rappel de la problématique :

"Le déséquilibre du coût métier entre un faux négatif (FN - mauvais client prédit bon client : donc crédit accordé et perte en capital) et un faux positif (FP - bon client prédit mauvais : donc refus crédit et manque à gagner en marge). Vous pourrez supposer, par exemple, que le coût d'un FN est dix fois supérieur au coût d'un FP."

Utilisation de la fonctionnalité **make_scorer** (ScikitLearn) :

cost = (fp + (10 * fn)) / len(y_true)



OBJECTIF : limiter le nombre de faux négatifs (FN)

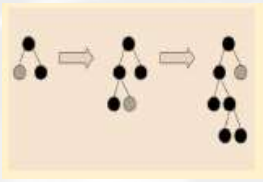
MODELISATION

Les modèles testés seront les suivants :

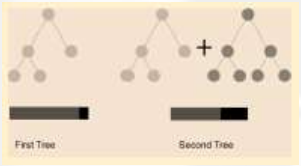
- RandomForestClassifier :

- XGBoostClassifier :

- LightGBM :



- CatBoostClassifier :

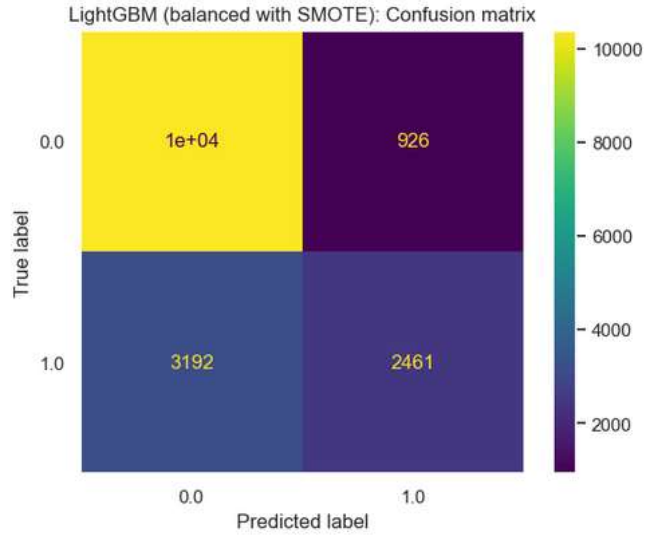


Les métriques d'évaluations seront les suivantes :

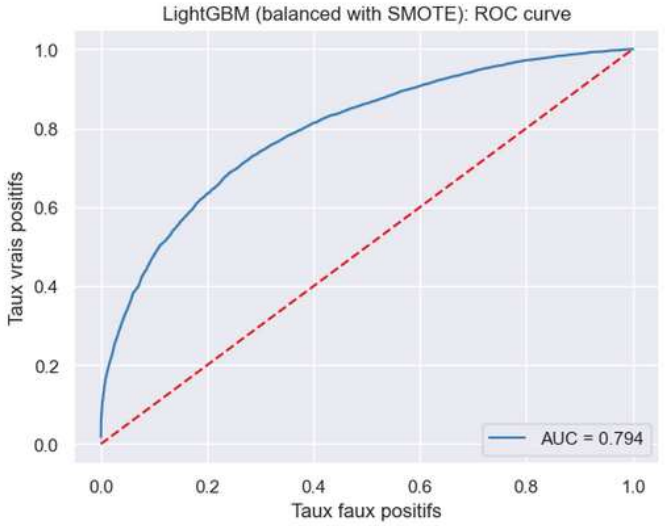
- AUC score : probabilité qu'un évenement soit classé comme positif par le test. Doit se rapprocher de 1.
- Precision : TP / (TP + FP)
- Recall : TP / (TP + FN)
- F-1 score : synthèse entre Precision et Recall.
  TP / (TP + ½(FN + FP))
- Accuracy : (TP + TN) / total
- Business score
- Temps d'entraînement : temps nécessaire au modèle pour s'entraîner et être capable d'effectuer des prédictions.

MODELISATION

Ci-dessous le résultat de la modélisation via le modèle LightGBM utilisant le rééquilibrage des données :




```
CPU times: total: 1min 1s
```



ANALYSE DES RESULTATS

Model	Business score	Time	Accuracy	Precision	Recall	F-1 score	AUC score
RandomForest (unbalanced)	0.806266	240.65	0.919271	0.709683	0.500643	0.480372	0.648676
RandomForest (SMOTE)	2.607229	44.60	0.695973	0.662036	0.580046	0.570085	0.689147
XGBoost (unbalanced)	0.771097	66.04	0.914491	0.625310	0.523121	0.525679	0.712149
XGBoost (SMOTE)	1.577098	20.36	0.722035	0.686362	0.683266	0.684711	0.751933
LightGBM (unbalanced)	0.796446	7.27	0.919580	0.748140	0.506782	0.493172	0.759487
LightGBM (SMOTE)	1.936789	2.73	0.757179	0.745706	0.676720	0.689471	0.794249
CatBoost (unbalanced)	0.772609	98.10	0.917808	0.672748	0.521894	0.522994	0.751449
CatBoost (SMOTE)	1.328085	49.18	0.715785	0.687462	0.699850	0.691402	0.772940

L'ATTENTION DOIT ICI ÊTRE PORTÉE SUR LE SCORE AUC, LE TEMPS ET LE RECALL AFIN DE MINIMISER LE NOMBRE DE CLIENTS DÉFAILLANTS PRÉDITS COMME NON DÉFAILLANTS.

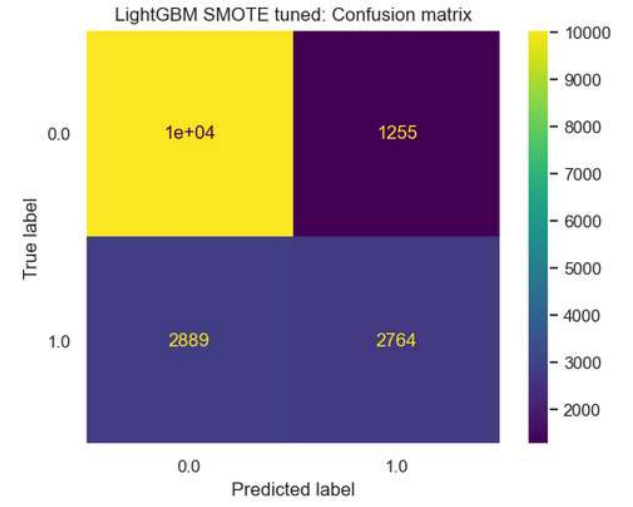
=> LE MEILLEUR MODÈLE EST LE LIGHTGBM (SMOTE)

OPTIMISATION

NOUS UTILISONS GRIDSEARCHCV SUR LE MODÈLE LIGHTGBM (SMOTE) AFIN D'OPTIMISER LE RÉSULTAT.

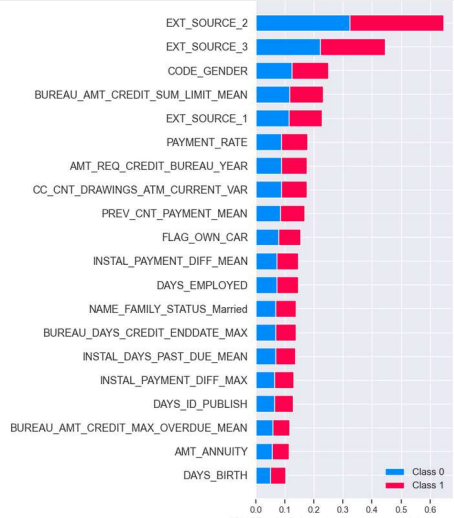
HYPER PARAMÈTRES UTILISÉS :

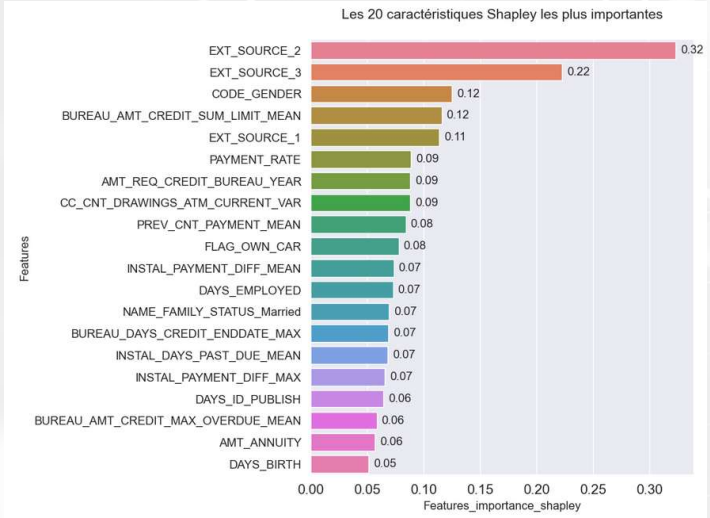
```
max_depth = [5, 10, 15]
num_leaves = [30, 40, 50]
learning_rate = [0.001, 0.01, 0.1]
subsample = [0.6, 0.7, 0.8]
colsample_bytree = [0.5, 0.55, 0.6]
```

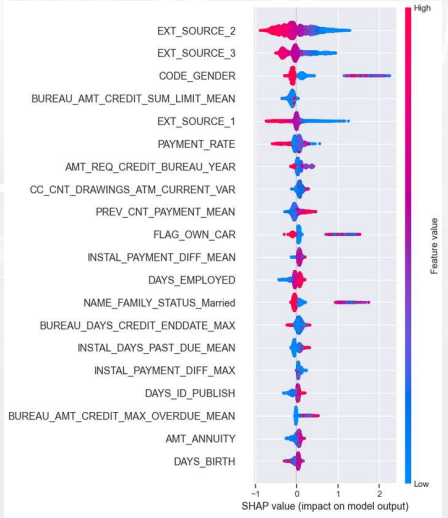


INTERPRETABILITE

ANALYSONS L'IMPORTANCE DES FEATURES VIA LA BIBLIOTHÈQUE SHAP







TESTS UNITAIRES

DES TESTS UNITAIRES SONT EFFECTUÉS VIA LA BIBLIOTHÈQUE PYTEST.

Ci-dessous un exemple d'exécution de test sur le code de génération du Dashboard :

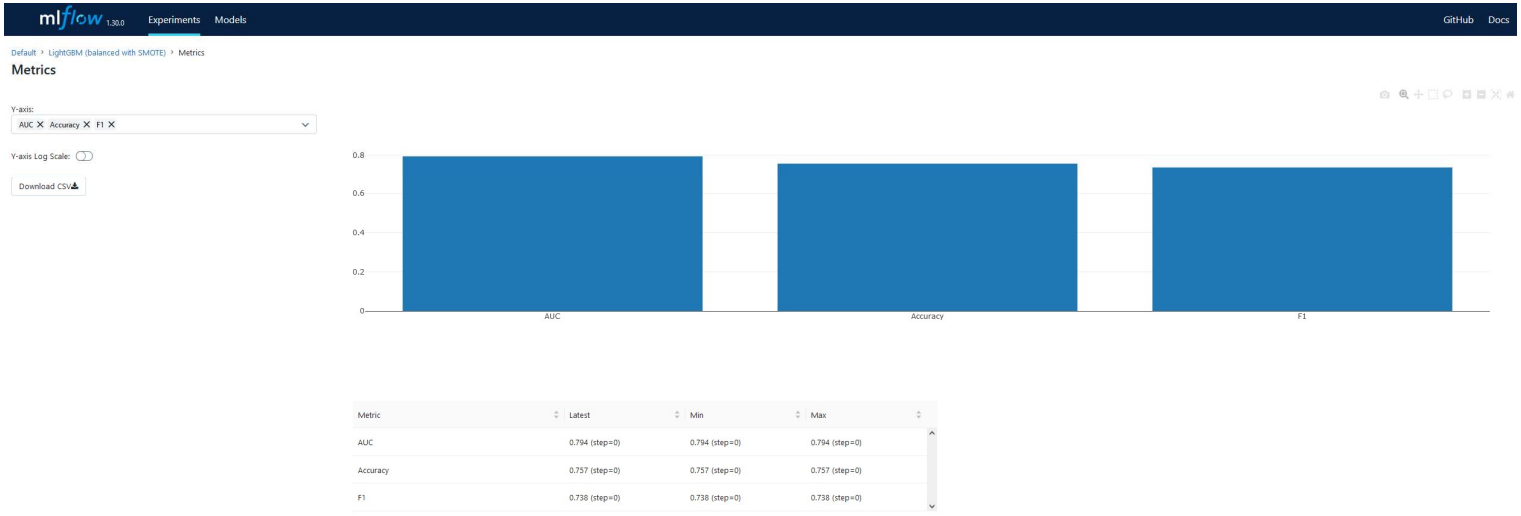
```
(EnvP6) C:\Users\JayCo\PROJET_7\tests>pytest
============================================================================= test session starts ==============================================================================
platform win32 -- Python 3.9.16, pytest-7.3.2, pluggy-1.0.0
rootdir: C:\Users\JayCo\PROJET_7\tests
plugins: anyio-3.5.0, dials-data-2.4.63
collected 3 items

test_streamlit_dashboard.py ...                                                                                                                                       [100%]

============================================================================= warnings summary =============================================================================
..\..\anaconda3\envs\EnvP6\lib\site-packages\shap\plots\_force.py:11
..\..\anaconda3\envs\EnvP6\lib\site-packages\shap\plots\_image.py:18
..\..\anaconda3\envs\EnvP6\lib\site-packages\shap\plots\_text.py:9
  Importing display from IPython.core.display is deprecated since IPython 7.14, please import from IPython display

test_streamlit_dashboard.py::test_gauge_function
  The get_cmap function was deprecated in Matplotlib 3.7 and will be removed two minor releases later. Use ``matplotlib.colormaps[name]`` or ``matplotlib.colormaps.get_cmap(obj)`` instead.

-- Docs: https://docs.pytest.org/en/stable/how-to/capture-warnings.html
========================================================================== 3 passed, 4 warnings in 4.12s ===========================================================================
```

Aucune erreur n'apparaît durant ce test.

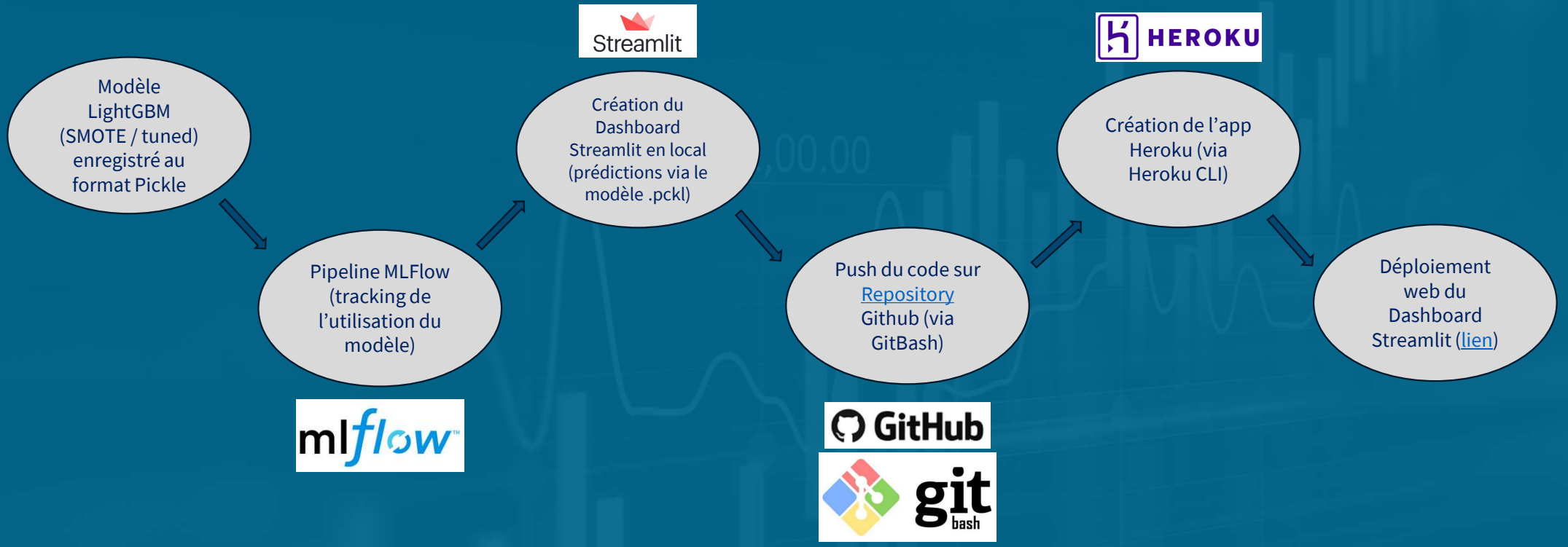
MLFLOW UI

VISUALISATION DU TRACKING/VERSIONING SUR MLFLOW UI :

Created	Duration	Run Name	User	Source	Version	Models	AUC	Accuracy	Custom score	max_depth	n_estimators	train_class_0
17 days ago	2.7s	LightGBM (balanced with SMOTE)	JayCo	C:\Users\...	-	sklearn	0.794	0.757	1.937	-1	100	45228
17 days ago	2.3s	XGBoost (balanced with SMOTE)	JayCo	C:\Users\...	-	sklearn	0.752	0.722	1.577	None	100	45228
17 days ago	2.7s	RandomForest (balanced with SMOTE)	JayCo	C:\Users\...	-	sklearn	0.689	0.696	2.607	None	100	45228
17 days ago	2.6s	LightGBM (unbalanced)	JayCo	C:\Users\...	-	sklearn	0.759	0.92	0.796	-1	100	226145
17 days ago	2.5s	XGBoost (unbalanced)	JayCo	C:\Users\...	-	sklearn	0.712	0.914	0.771	None	100	226145
17 days ago	3.9s	RandomForest (unbalanced)	JayCo	C:\Users\...	-	sklearn	0.649	0.919	0.806	None	100	226145



PIPELINE DE DEPLOIEMENT



ANALYSE DU DATADRIFT

Rapport de Data Drift créé selon la blibliothèque Evidently :

Dataset Drift

Dataset Drift is NOT detected. Dataset drift detection threshold is 0.5

17	4	0.235
Columns	Drifted Columns	Share of Drifted Columns

Data Drift Summary

Drift is detected for 23.529% of columns (4 out of 17).

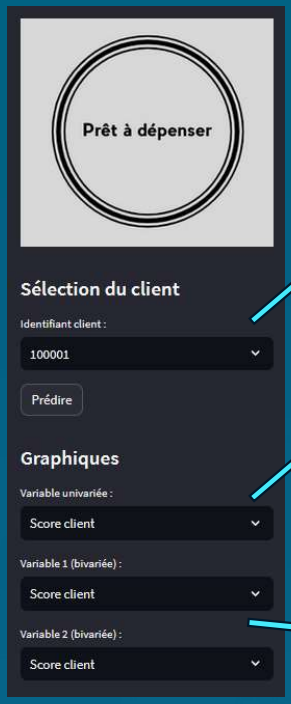
Search

Column	Type	Reference Distribution	Current Distribution	Data Drift	Stat Test	Drift Score
BUREAU_MONTHS_BALANCE_SIZE_SUM	num			Detected	Wasserstein distance (normed)	1.38934
AMT_REQ_CREDIT_BUREAU_QRT	num			Detected	Wasserstein distance (normed)	0.359036
AMT_CREDIT	num			Detected	Wasserstein distance (normed)	0.207392
NAME_CONTRACT_TYPE	num			Detected	Jensen-Shannon distance	0.147536
EXT_SOURCE_2	num			Not Detected	Wasserstein distance (normed)	0.049562
PREV_DAYS_DECISION_MEAN	num			Not Detected	Wasserstein distance (normed)	0.048603

ANALYSE DU DATADRIFT

L’exemple ci-dessous de Data Drift est provoqué par le déséquilibre de classe entre les datasets d’entraînement et de test.

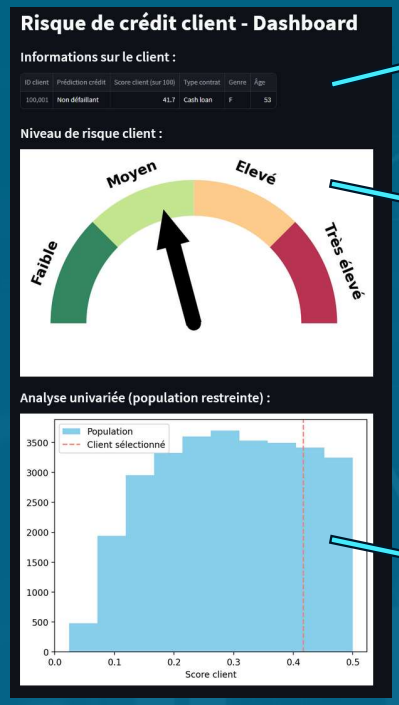
PRESENTATION DU DASHBOARD



Sélection du client parmis la liste du fichier 'test'

Sélection des features pour le graphique univarié

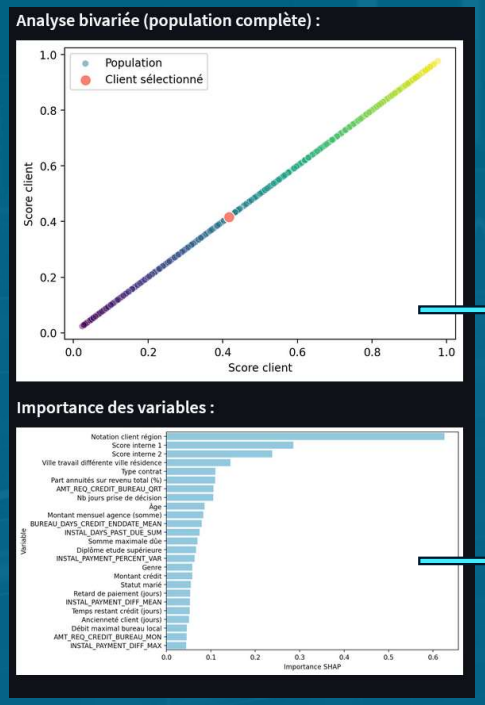
Sélection des features pour le graphique bivarié



Informations relatives au client sélectionné

Jauge du niveau de risque du client

Graphique univarié (barplot)



Graphique bivarié (scatterplot)

Top25 des features les plus importantes

PRESENTATION DU DASHBOARD

Démo de l’utilisation du Dashboard intéractif :

Lien

Axes d'amélioration

- Optimisation de la fonction coût métier en collaboration directe avec le métier
- Décomposition des variables de scoring interne ('EXT_SOURCE'), qui présentent une importance relative dans la modélisation
- Approfondir la création de nouvelles variables

- Optimisation du score AUC (meilleur score Kaggle = 0,82)
- Optimiser l'encodage et l'imputation des données
- Approfondir l'optimisation des hyper-paramètres
- Ajouter un système d'authentification pour les utilisateurs afin de sécuriser l'utilisation du dashboard intéractif
- Encrypter les données clients dans un souci de confidentialité

SYNTHÈSE

MERCI