OPENCLASSROOM

Parcours data scientist en alternance *(@Synapse-Medicine)*

Implémentez un modèle de scoring

16 décembre 2022

Maxime Dupouy

Livrable 7 Implémentez un modèle de scoring

Introduction

Objectifs du livrable

Présentation BDD et features selection/engineering

2 - Modélisation

Méthodologie

Résultats

4 - Dashboard

Structuration

Démostration

1 - Fonction coût-métier

3 - Interprétabilité

Conclusion

Introduction

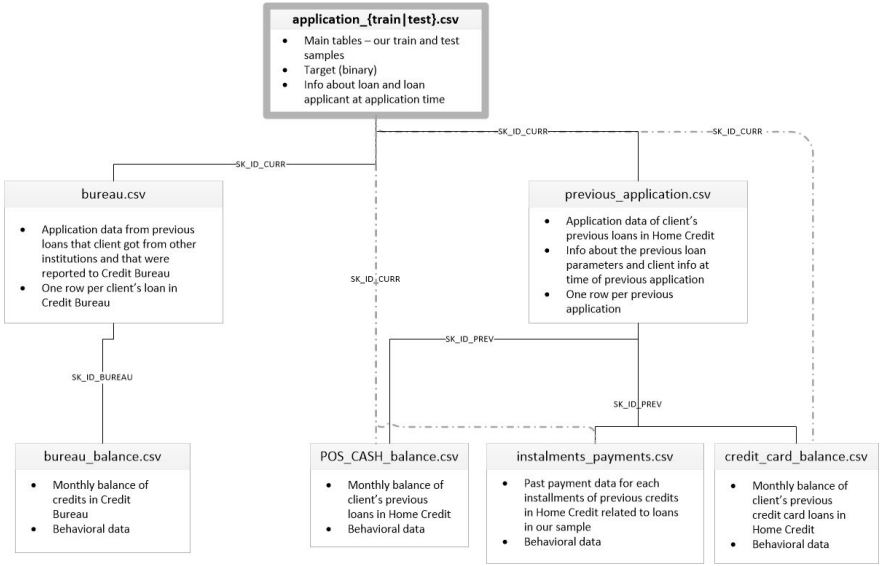
Objectifs

Implémentez un modèle de scoring

Missions	Objectifs
- Construire un **modèle de scoring** qui donnera une **prédiction** sur la probabilité de faillite d'un client de façon automatique. - Construire un **dashboard interactif** permettant d'**interpréter les prédictions** faites par le modèle, et d'améliorer la connaissance client des chargés de relation client.	- Présenter son travail de modélisation à l'oral - Réaliser un dashboard - Rédiger une note méthodologique - Utiliser un logiciel de version de code - Déployer un modèle via une API dans le Web

Introduction

BDD

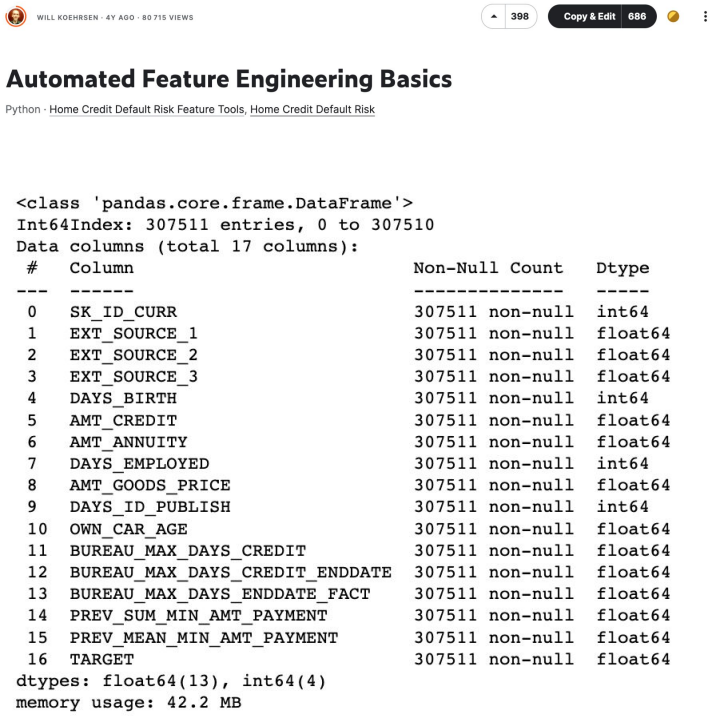


Donnée provient de Home Credit Bank et est hébergée sur Kaggle.

Le but de Home Credit est d'élargir l'inclusion financière de la population non bancarisée.
Ils utilisent une variété de données alternatives, comme des informations sur les télécommunications et les transactions, pour prédire les capacités de remboursement de ses clients.

Introduction

Features selection/engineering



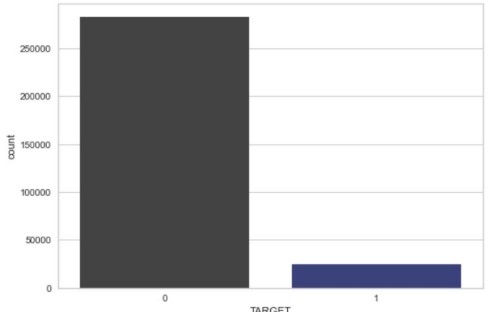
Partie “sous traitée” sur des notebooks déjà réalisés par des data scientist - lien

Obtention d’une BDD composée de :

- 15 colonnes les plus “impactantes” (sélection avec *“Feature Tool”* et *“Deep Feature Synthesis”*)
- 307 511 observations
- TARGET :
  - 0 = client sans défaut de paiement
  - 1 = client en défaut de paiement

1 - Fonction coût-métier

Problématique et résolution



Répartition des classes 0 et 1

	Prédiction clients non défaut (0)	Prédiction clients en défaut (1)
Clients non défaut (0)	**TN**	**FP**
Clients en défaut (1)	**FN**	**TP**

Matrice de confusion

Objectifs:

Éviter de se tromper lorsque le client est à défaut de paiement, tout en acceptant le plus de crédits possible.

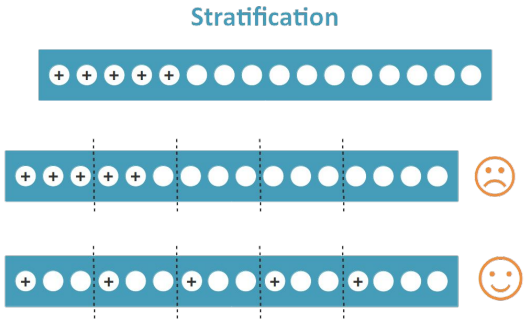
Métriques sélectionnées :

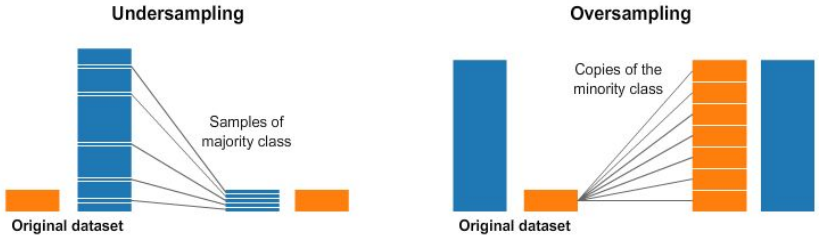
- recall
- précision

Une métrique existe pour avoir un ratio entre les deux : le F ß 2 score.

$$F_\beta = (1 + \beta^2) \cdot \frac{precision \cdot recall}{(\beta^2 \cdot precision) + recall}$$

2 - Modélisation
Méthodologie





Méthode pour évaluer de manière robuste :

- Stratified K fold

Contrebalancer le déséquilibre des classes :

- Oversampling (SMOTE)
- Undersampling (NearMiss)
- Variation threshold

Modèles entraînés:

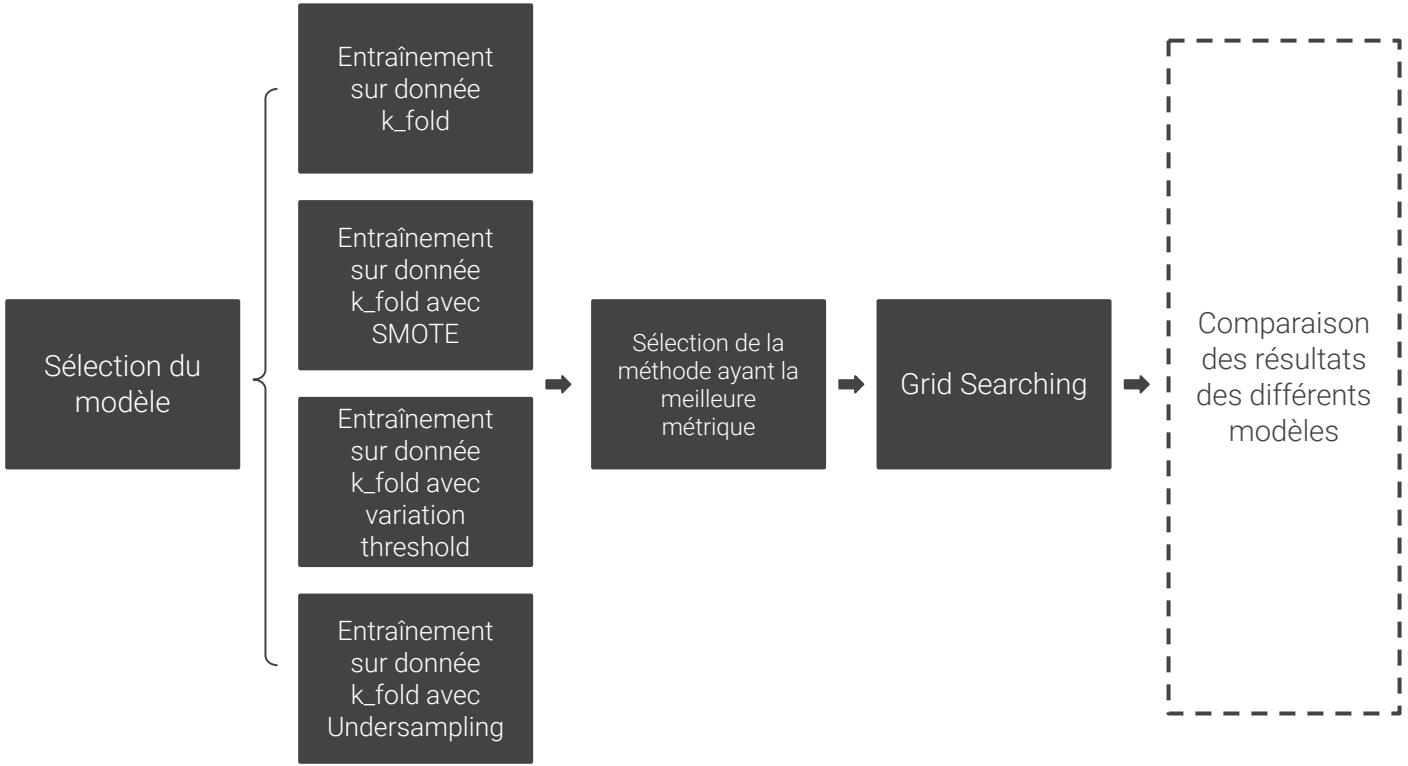
- Logit
- RandomForestClassifier base
- RandomForestClassifier notebook
- XGBoostClassifier
- LGBMClassifier

Evaluation sur :

- F ß 2 score
- recall, précision, accuracy

2 - Modélisation

Méthodologie



Résumé

2 - Modélisation

Résultats

Baseline :

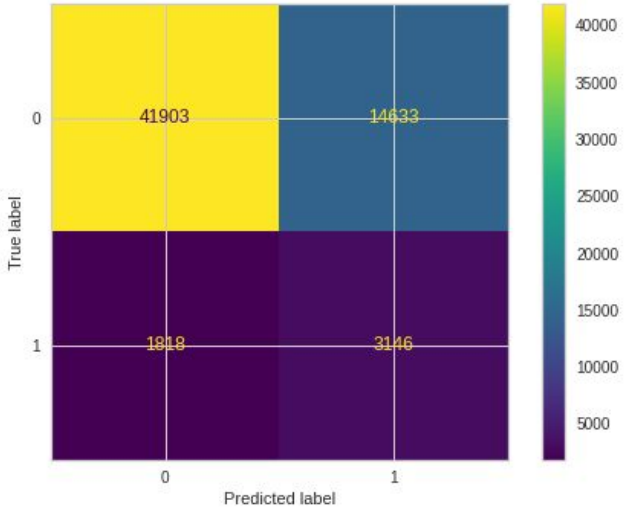
- F ß 2 = 0.086
- accuracy = 0.85

	F ß2 score	**Accuracy**
Logit (SMOTE)	0.39	0.68
RandomForestClassifier (finetuné)	0.26	0.33
RandomForestClassifier finetuné notebook	0.399	0.68
LightGBM non finetuné	0.42	0.69
XGBoost finetuné	0.418	0.73

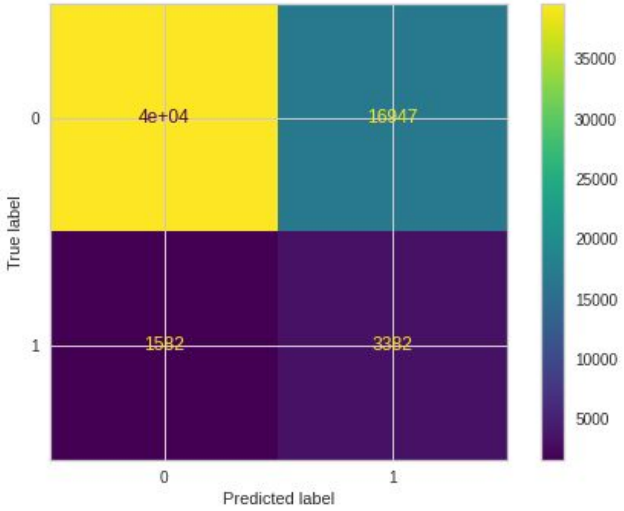
Benchmark

2 - Modélisation

Résultats



LightGBM

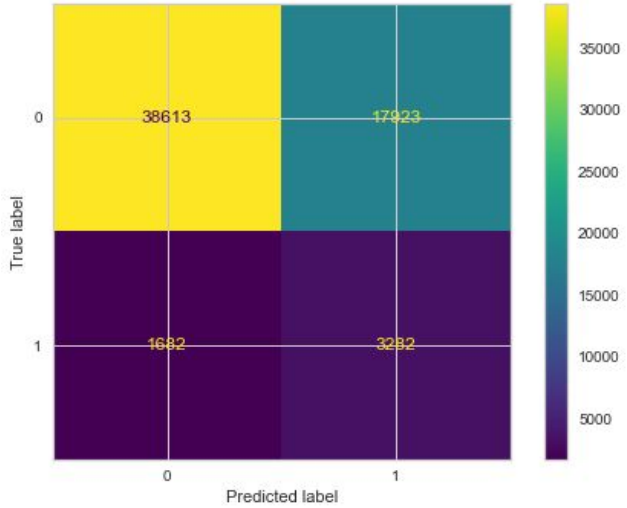


XGBoost

Matrice de confusion

2 - Modélisation

Choix du modèle



	random_forest_notebook
f_beta_score_	0.399685
precision_	0.681218
recall_	0.661148
accuracy_	0.681218

Implémentation du **RandomForestClassifier** notebook, sans oversampling, undersampling, nouveau finetuning.

Choix sur F ß 2 score **et** recall

3 - Interprétabilité

Globale

Réalisée avec scikit learn (feature importance)

Locale

Réalisée avec Shap (Force plot)

Descriptif des clients

Box plot afin de visualiser où se situe le client par rapport à la distribution globale

Implémentation

4 - Dashboard

FastAPI

- gère la data
- gère modèle ML
- génère graphique

Streamlit

- gère affichage
- gère input de données clients

heroku

GitHub

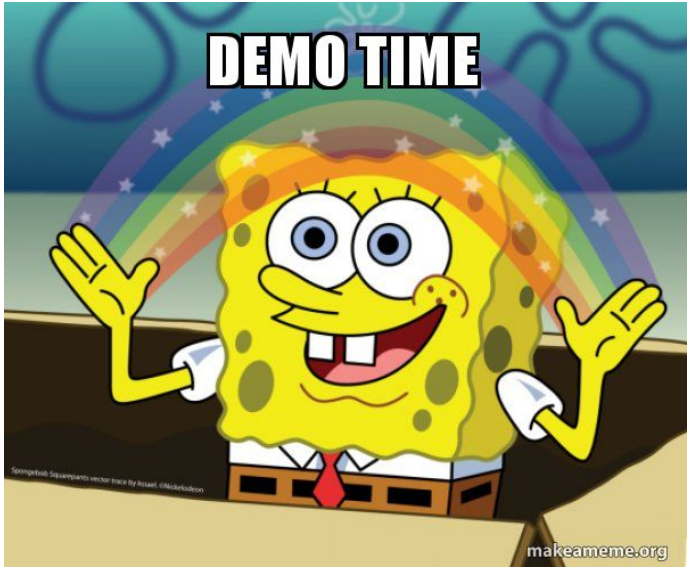
Backend → **Fastapi**

Frontend → **Streamlit**

Déploiement → **Heroku**

Versionning → **Github**

4 - Dashboard



🤞🤞🤞🤞

Conclusion

https://www.cogitime.fr/vis-ma-vie-de-developpeur-web/

Modélisation :

- Résultats supérieurs à la baseline, mais reste assez faibles pour une utilisation brute
- Axes d'amélioration : réaliser la feature selection/engineering, tester modèle bagging, finetuner sur sample_weight

Dashboard :

- Répond au cahier des charges
- Axes amélioration : performance du site mitigée, graphiques interactifs

Global :

- S'appuyer sur des connaissances métiers et/ou de la donnée afin de réaliser certains choix

Points de difficultés :

- Ne pas avoir fait la feature selection : gain de temps mais mauvaise compréhension de la donnée
- Travail sur mac m1 (puce silicone) : problème de compatibilité (Heroku, xgboost, lgbm)

Je vous remercie pour votre attention

Dashboard

Versioning

GitHub

Désolé si le premier livrable ne correspond pas à ce que je présente, je ne comprends pas trop la consigne …
Le site sera "destroy" à la fin de la soutenance pour des raisons évidentes de coût