Implémentez un modèle de scoring

Projet 7 Parcours Data Scientist
Python Jupyter Notebook - Google Colab
Github > Streamlit > Heroku

Nicolas Pasero Mai 2021

OpenClassrooms - Centrale Supélec

Sommaire

1. Rappel du contexte et problématique

2. Présentation des données

3. Analyse exploratoire des données

4. Modélisation

5. Présentation du dashboard

Rappel du contexte Problématique

Décision d'octroi de crédit…

Contexte et problématique

Entreprise "Prêt à dépenser"

Crédits à la consommation pour des personnes ayant peu d'historique de prêt.

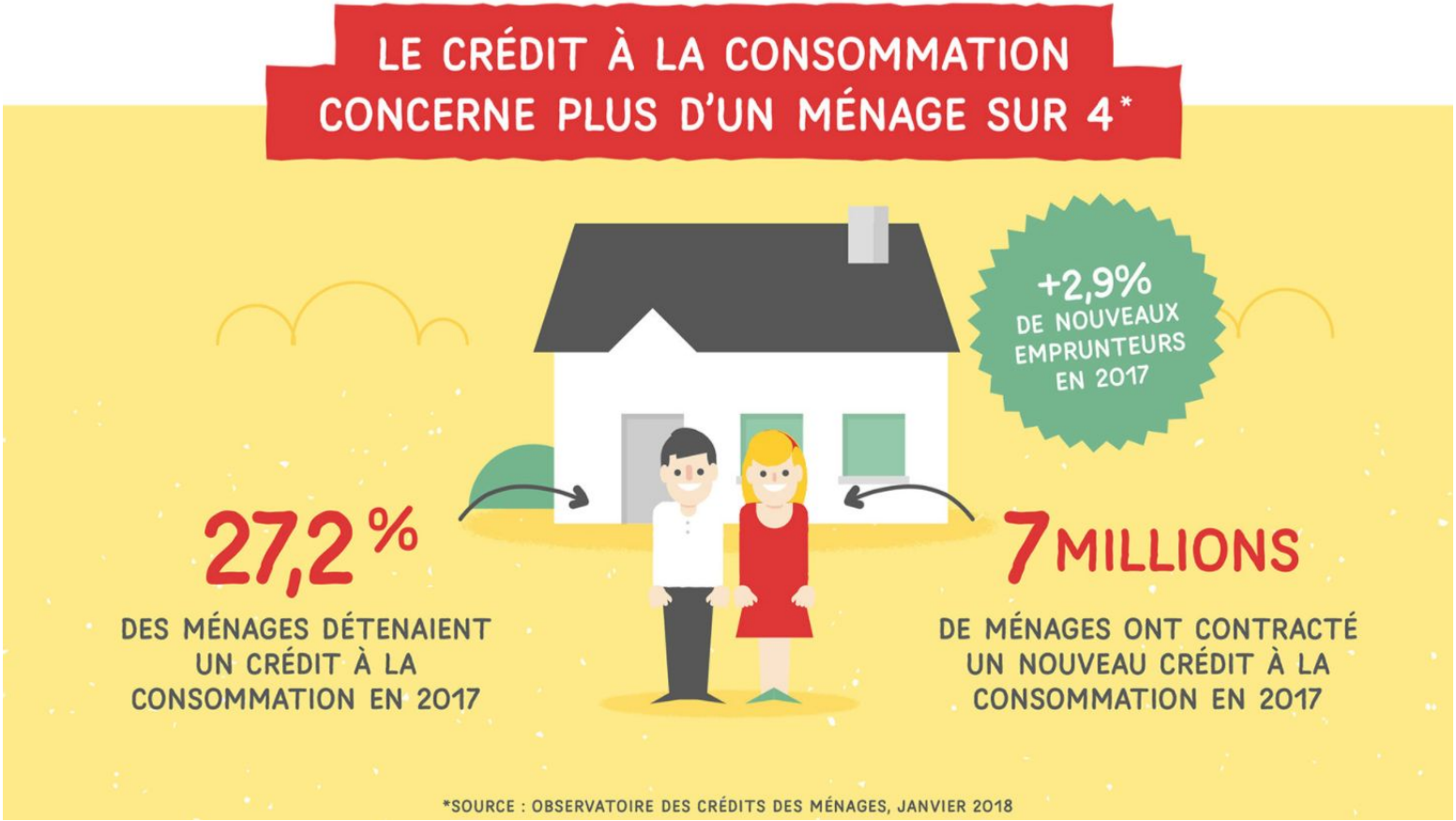
Besoin

Modèle de scoring de la probabilité de défaut de paiement du client.

Objectif

Dashboard interactif à destination des chargés de relation client.

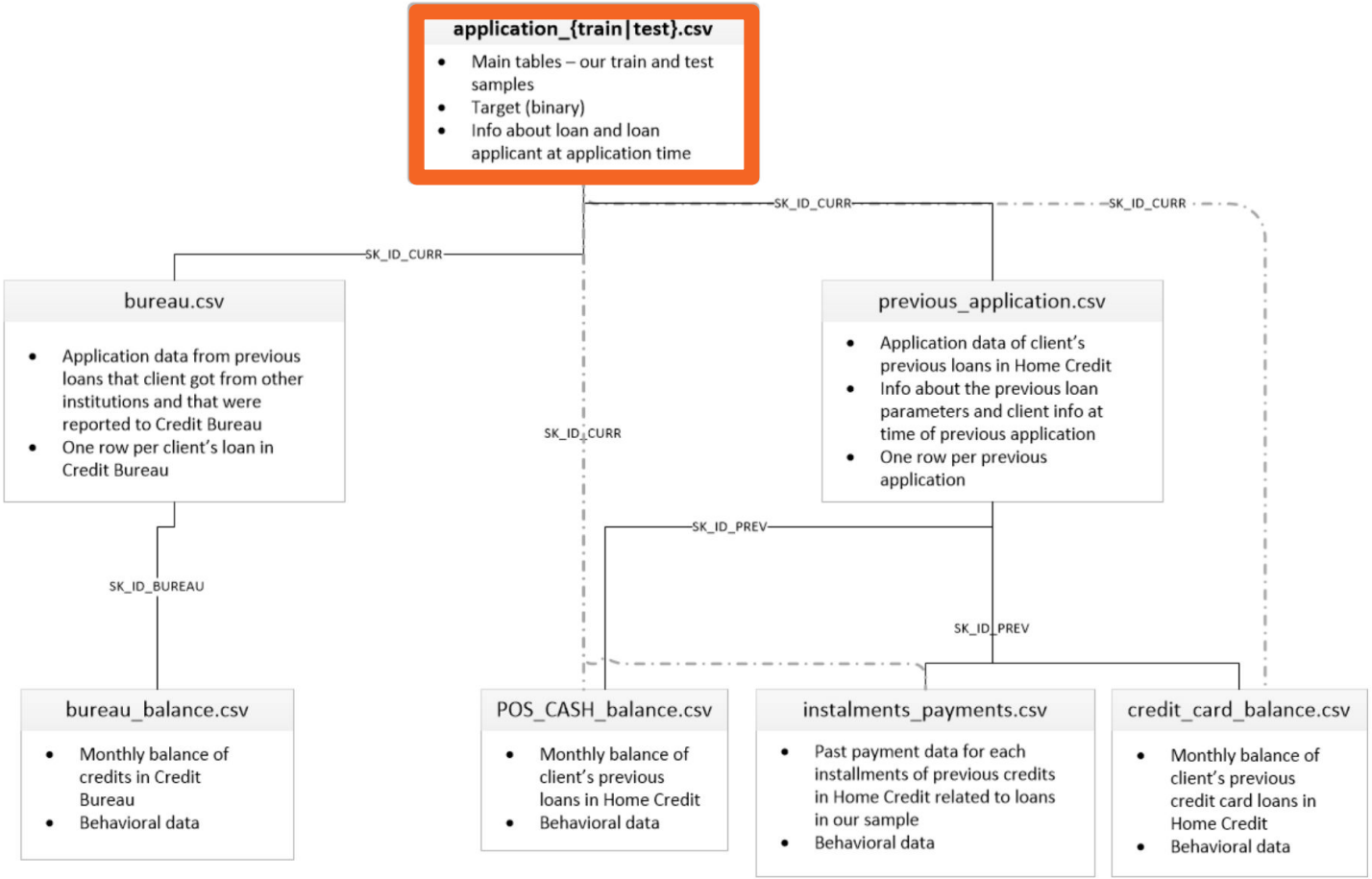
Impact marché !



Présentation des données

Kaggle "Home Credit Default Risk" :
https://www.kaggle.com/c/home-credit-default-risk/data

Schéma de la BDD *source Kaggle*



Description rapide des données

	Rows	Columns	%NaN	%Duplicate	object_dtype	float_dtype	int_dtype	bool_dtype	MB_Memory
./data/application_test.csv	48744	121	23.81	0.0	16	65	40	0	44.998
./data/POS_CASH_balance.csv	10001358	8	0.07	0.0	1	2	5	0	610.435
./data/credit_card_balance.csv	3840312	23	6.65	0.0	1	15	7	0	673.883
./data/installments_payments.csv	13605401	8	0.01	0.0	0	5	3	0	830.408
./data/application_train.csv	307511	122	24.40	0.0	16	65	41	0	286.227
./data/bureau.csv	1716428	17	13.50	0.0	3	8	6	0	222.620
./data/previous_application.csv	1670214	37	17.98	0.0	16	15	6	0	471.481
./data/bureau_balance.csv	27299925	3	0.00	0.0	1	0	2	0	624.846
./data/sample_submission.csv	48744	2	0.00	0.0	0	1	1	0	0.744

Jeu de données principal

Analyse exploratoire des données

Inspiré par le Kernel :
https://www.kaggle.com/willkoehrsen/start-here-a-gentle-introduction
https://www.kaggle.com/danilz/merge-all-data-base-glm-vs-xgb-explained-0-763

Preprocessing

Identification / imputation des **valeurs manquantes**.

Analyse des **outliers** / valeurs atypiques.

Visualisation des **corrélations** avec notre cible.

Encodage des variables catégorielles.

Standardisation des données.

Opération de Merging

Enrichissement de l'échantillon de travail :

Combinaison des **7 jeux de données**.

Avant 123 features - Après **189 features**

dont 3 features de moyenne et de comptage :

PREVIOUS_LOANS_COUNT : nombre des précédents crédits pris par le client

MONTHS_BALANCE_MEAN : solde mensuel moyen des précédents crédits

PREVIOUS_APPLICATION_COUNT : nombre de demandes antérieures au crédit immobilier

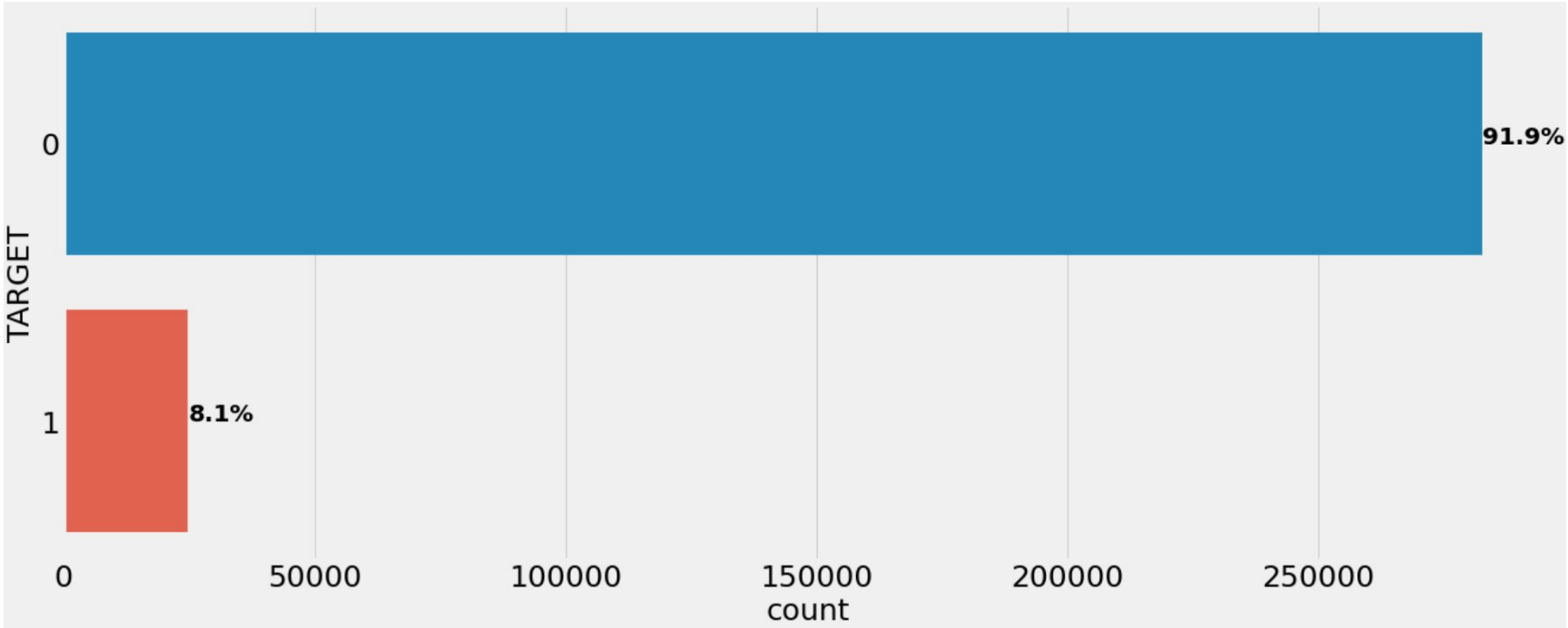
Feature engineering

Enrichissement de l'échantillon par **4 ratios explicatifs** :

CREDIT_INCOME_PERCENT : % montant du crédit par rapport au revenu d'un client
ANNUITY_INCOME_PERCENT : % rente de prêt par rapport au revenu d'un client
DAYS_EMPLOYED_PERCENT : % jours employés par rapport à l'âge du client
CREDIT_TERM : durée du paiement en mois

Echantillon de travail obtenu : 356255 x 193

Distribution de la cible



Oversampling >>> SMOTE

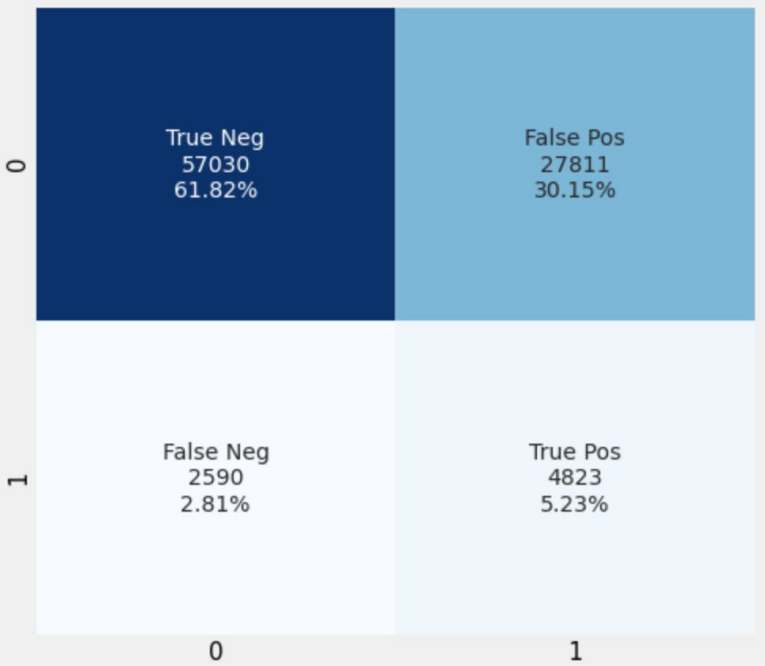
Technique utilisée pour traiter des ensembles de **données déséquilibrées**.

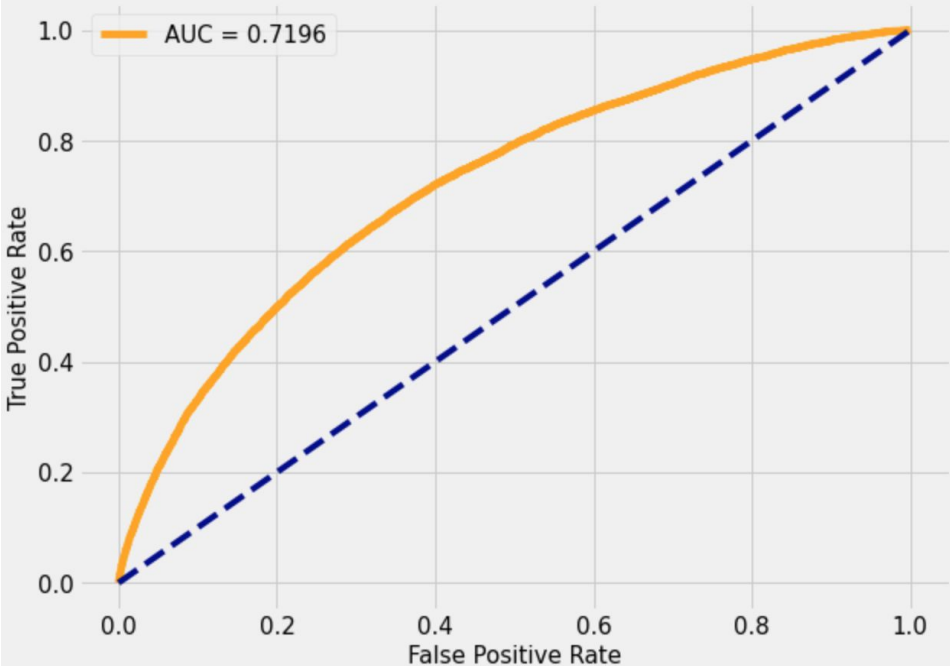
Modélisation

Baseline fixée par régression logistique

Elaboration d'un modèle, optimisation et compréhension.

Baseline Régression logistique





Performances de la "baseline" avec toutes les features.

Approche Gradient Boosting

Boosting de Gradient algorithme d'apprentissage supervisé.
Combiner les résultats d'un ensemble de modèles simples.
Principe d'auto-amélioration séquentielle.

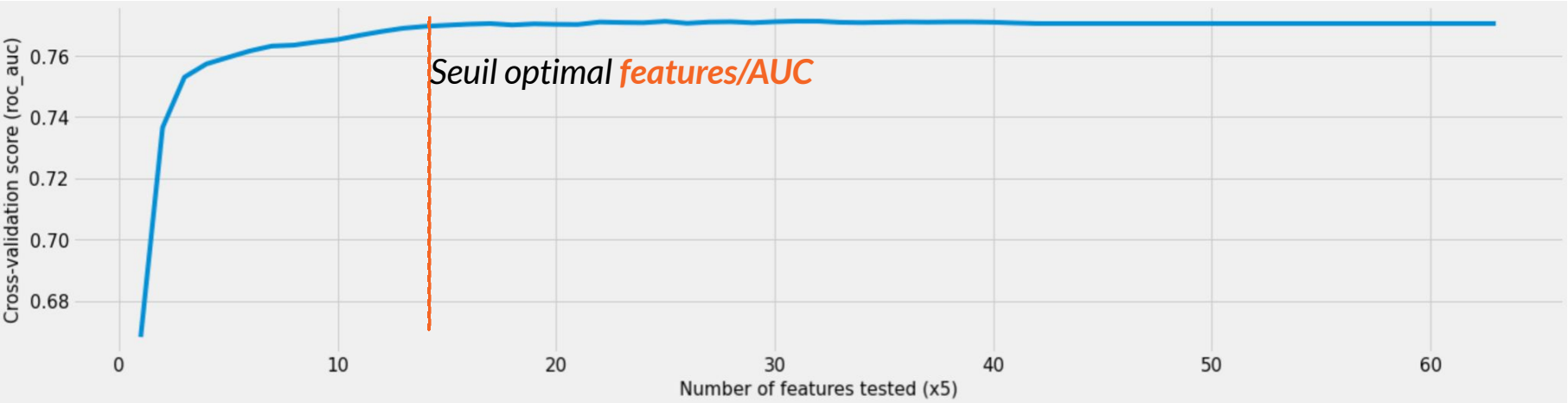
	Model	AUC	Accuracy	Precision	Recall	F1	Time
1	LGBMClassifier	0.773782	0.920264	0.563758	0.0339943	0.0641221	10.7289
0	CatBoostClassifier	0.773559	0.920144	0.5625	0.0279239	0.0532065	133.957
2	XGBClassifier	0.764208	0.919841	0.59375	0.00768919	0.0151818	4.2202

*Entraînement avec un **GPU** Google Colab.*

Feature selection

Recursive Feature Elimination : RFECV

Identification des best features par validation croisée en optimisant la métrique AUC.



Recursive elimination ⟹ *149 features*

Fonction coût

Limiter les risques de perte financière :

Pénaliser les *Faux Positifs* et les *Faux Négatifs*.

Quantification de l'importance relative entre *Recall* et *Precision*.

Estimation du coût moyen d'un défaut de paiement.

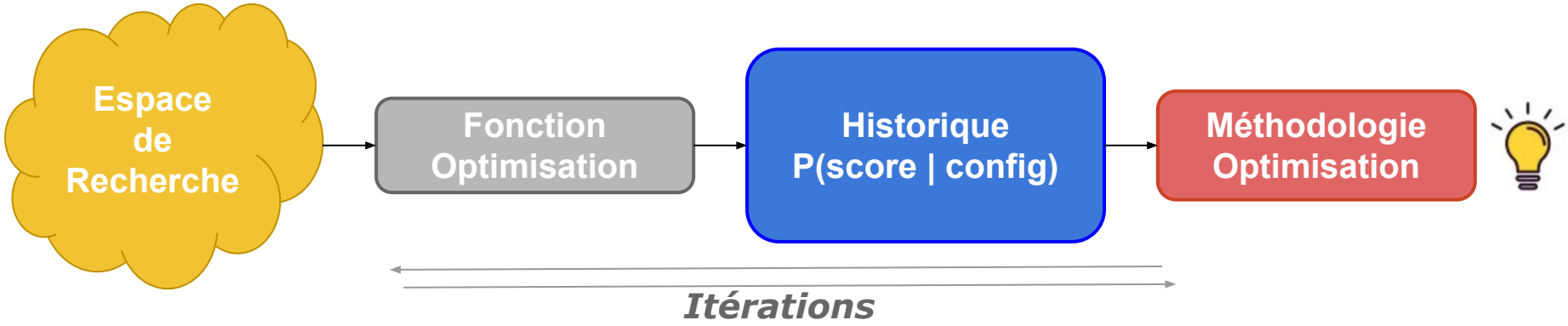
Estimation du coût d'opportunité d'un client refusé par erreur.

Connaissance métier nécessaire ou hypothèses à fixer.

Optimisation des Hyperparamètres

Choix d'une méthode avancée : HyperOpt

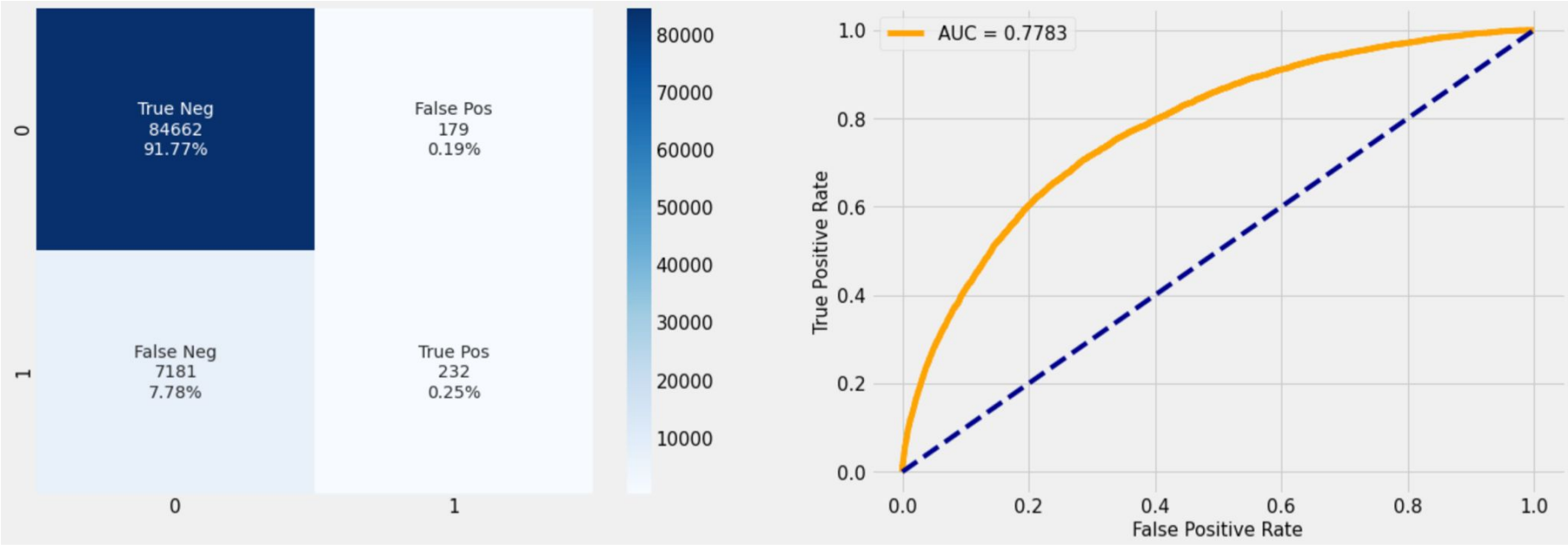
Automatisation de la recherche d'une configuration optimale d'hyperparamètres.
Solution basée sur l'**optimisation bayésienne**.



Pipeline d'optimisation ML HyperOpt.

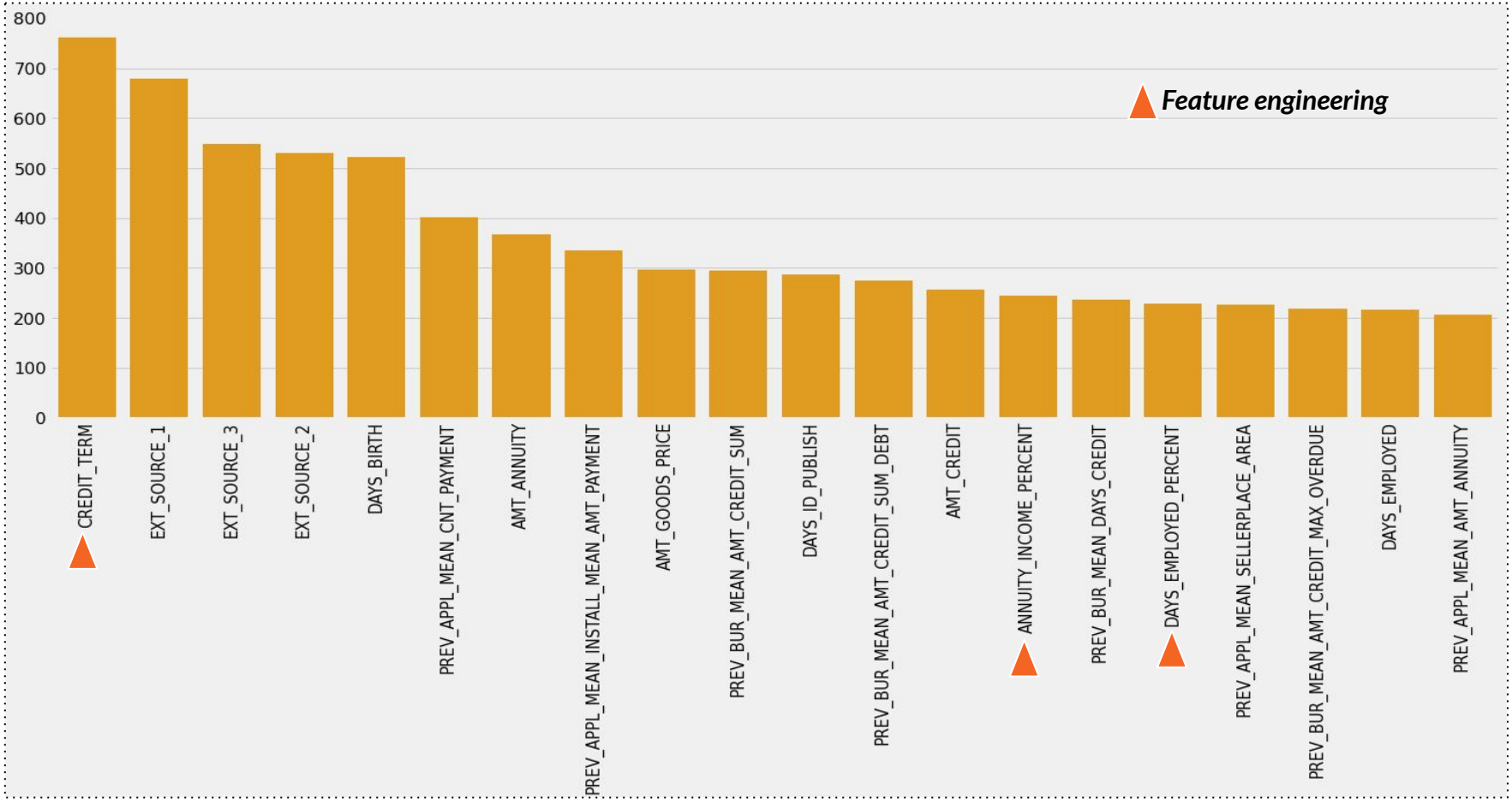
Métrique d'évaluation

Amélioration de la métrique et **pénalisation** des erreurs FP et FN.
Meilleures performances **LightGBM**.



Matrice de confusion + Courbe ROC / AUC score.

Feature importance "Top 20"



Présentation du dashboard

Versioning Github :
https://github.com/nalron/project_credit_scoring_model
Application : https://bank-credit-risk.herokuapp.com/

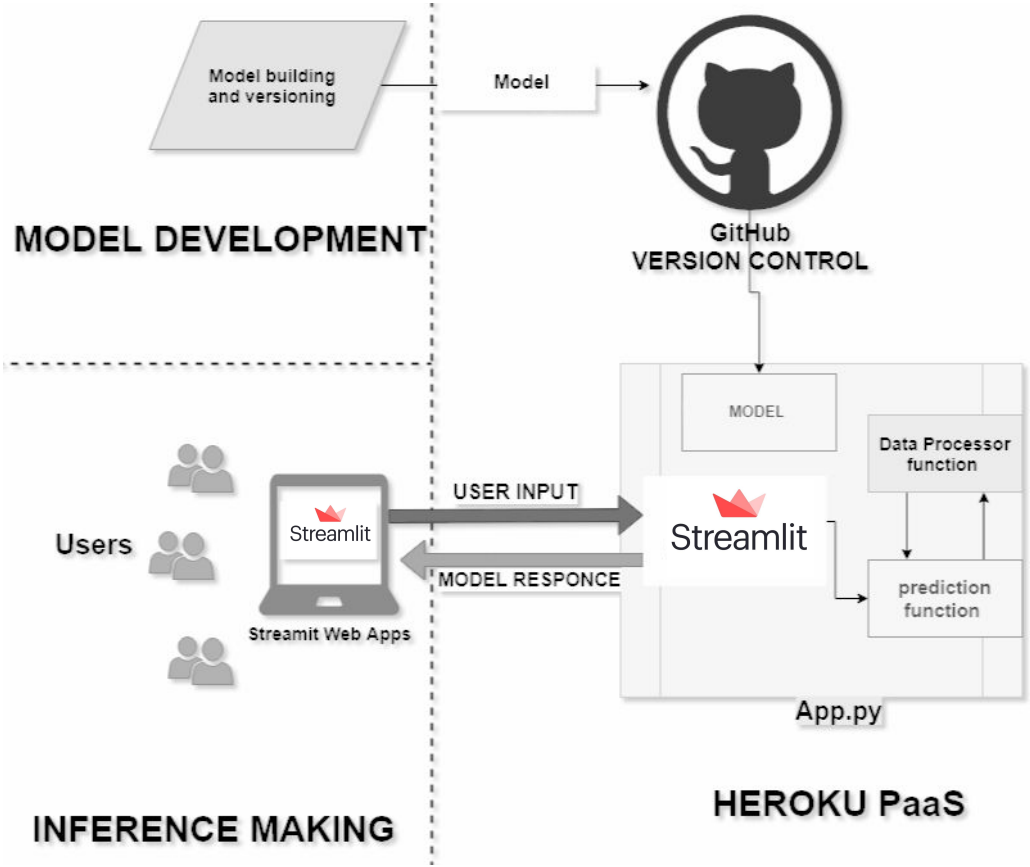
Streamlit

Streamlit : framework open-source Python spécialisé ML.
Application web : modèles de ML et visualisation des données.
Application performante : mise en cache via une annotation.
Création facile : sans nécessité d'implémenter du HTML.

	Maturity	Popularity	Simplicity	Adaptability	Focus	Language support
Streamlit	C	A	A	C	Dashboard	Python
Dash	B	A	B	B	Dashboard	Python, R, Julia
Panel	C	B	B	B	Dashboard	Python
Shiny	A	B	B	B	Dashboard	R
Voila	C	C	A	C	Dashboard	Python, R, Julia
Jupyter	A	A	B	B	Notebook	Python, R, Julia
Flask	A	A	B	A	Web framework	Python

source : www.datarevenue.com

Schéma fonctionnel de l'application



Conclusion

Utilisation et modification d'un Kernel Kaggle.

Entraînement d'un modèle de scoring.

Fonction coût, optimisation et évaluation.

Interprétabilité du modèle LightGBM.

Dashboard interactif.

lien vers le dashboard

share.streamlit.io