

Implémentez un modèle de scoring

Data & Analytics
Eric Blanvillain - 07-03-2022



INTRODUCTION

Introduction - Problématique



Mission :

- Développer un modèle de scoring de la probabilité de défaut de remboursement du client
- Développer un dashboard interactif

Objectifs :

- Aider à la décision d'accorder ou non un prêt à un client potentiel
- Expliquer de façon la plus transparente possible les décisions d'octroi de crédit
- Permettre aux clients de disposer des informations les plus pertinentes

EDA

Analyse de la
donnée initiale

CONSTRUCTION DU MODÈLE

Méthodologie
D'entraînement du
modèle
Performance du modèle

NOTE MÉTHODOLOGIQUE

Rédaction de la note
méthodologique du
modèle

DASHBOARD

Réalisation du
dashboard
Sauvegarde sur un
dépôt GitHub

DÉPLOIEMENT

Déploiement du modèle
via share.streamlit.io
[https://share.streamlit.io](https://share.streamlit.io/dashboard.py)
/dashboard.py


Introduction - Compétition Kaggle

kaggle

Featured Prediction Competition

Home Credit Default Risk

Can you predict how capable each applicant is of repaying a loan?

 Home Credit Group · 7,176 teams · 3 years ago

\$70,000
Prize Money

OverviewDataCodeDiscussionLeaderboardRules









Late Submission...

[Public Leaderboard](#)[Private Leaderboard](#)

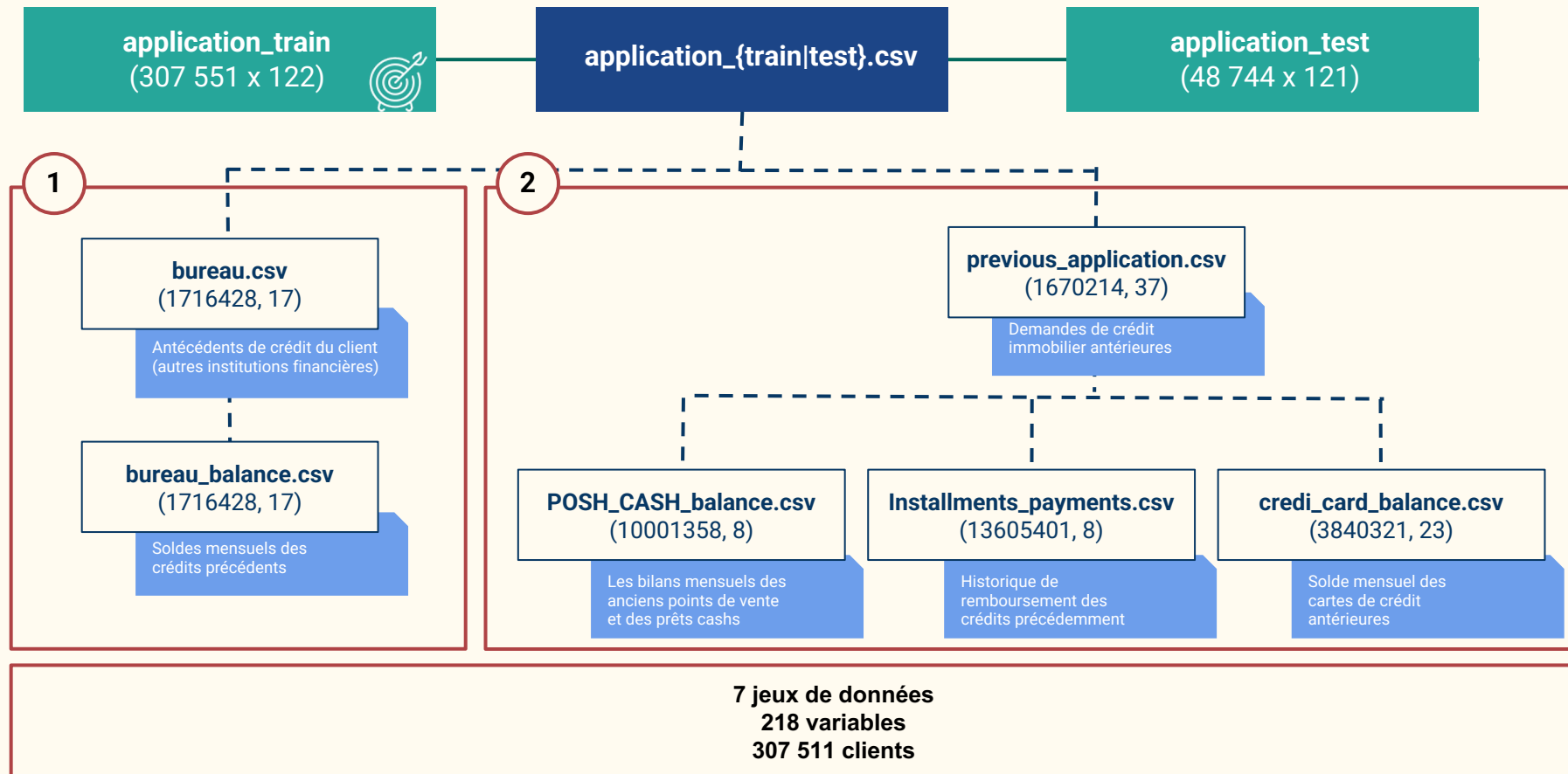
The private leaderboard is calculated with approximately 80% of the test data.
This competition has completed. This leaderboard reflects the final standings.

Refresh

In the moneyGoldSilverBronze

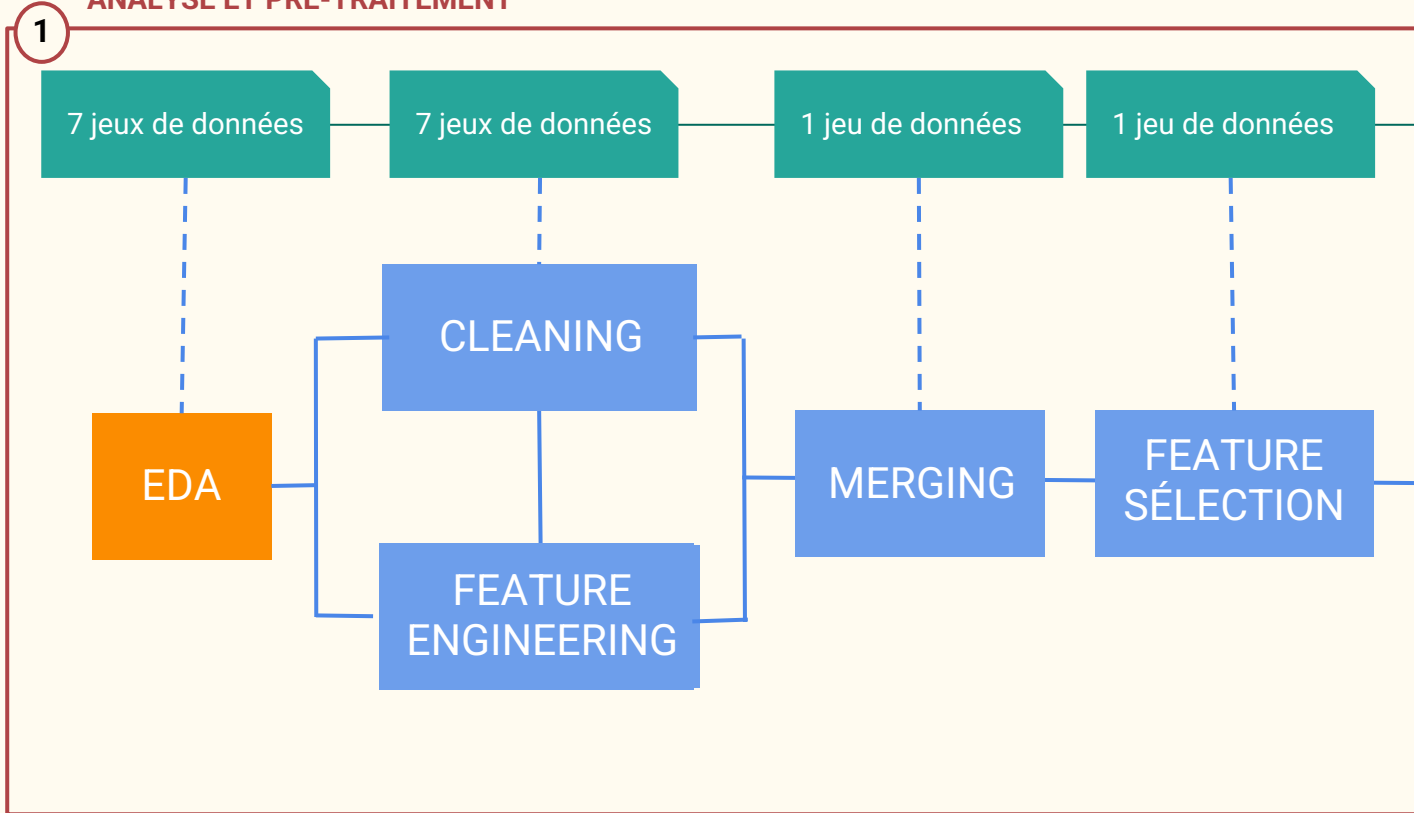
#	Δpub	Team Name	Notebook	Team Members	Score ?	Entries	Last
1	▲10	Home Aloan		   +3	0.80570	499	3Y
2	—	ikiri_DS		   +9	0.80561	477	3Y
3	▲1	alijs & Evgeny		 	0.80511	143	3Y

Introduction - Présentation de la donnée

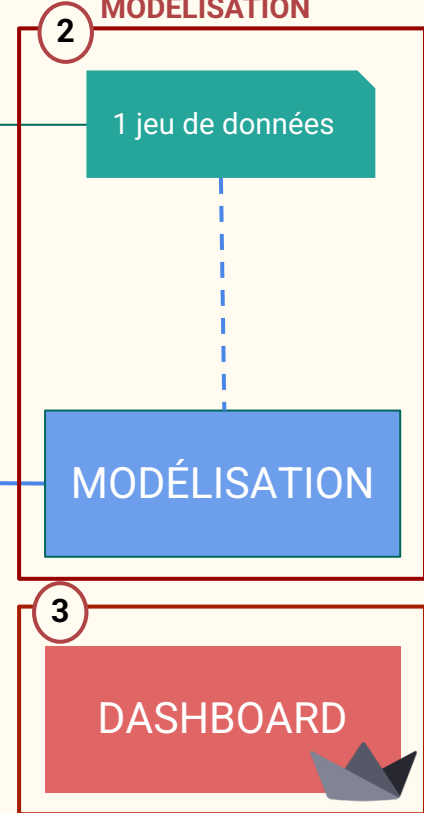


Introduction - Structure du projet

1 ANALYSE ET PRÉ-TRAITEMENT



2 MODÉLISATION



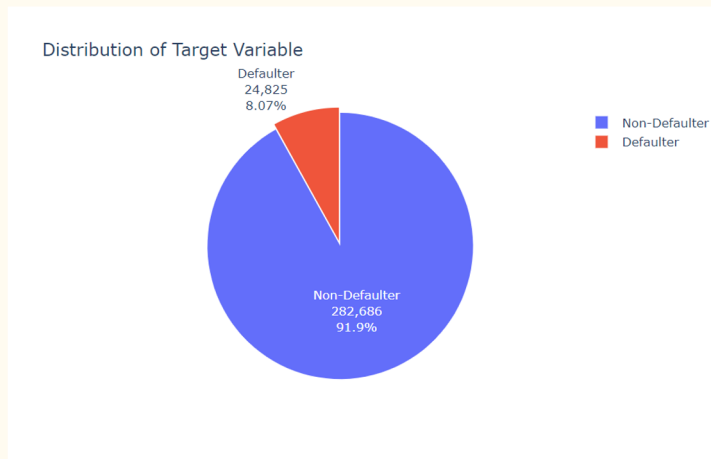
3

DASHBOARD



PART I
EXPLORATION ET ANALYSE INITIALE

Exploration initiale : variable cible (TARGET)



Donnée binaire (0 et 1)

Défaillants : 1

Non défaillants : 0

Déséquilibre des données de part la distribution de la variable **TARGET**

Défaillants : 8,07%, classe minoritaire

Non défaillants : 91,9%, classe majoritaire

-> Nous ne pouvons pas fournir les données telles quelles à nos algorithmes, qui peuvent être sensibles au déséquilibre des données

-> Nous devons utiliser des mesures telles que **le score ROC-AUC, la perte de log, le score F1, la matrice de confusion** pour une meilleure évaluation du modèle

Exploration initiale : process pour chaque dataset

EDA

1/3 variables : NaN > 50%

ERREURS*

Corrections
des aberrations
détectées

VALEURS
MANQUANTES*

Suppression
des colonnes
NaN > 60%

IMPUTATION*

Stratégie ≠
selon type

Qualitative

constante
(mode)

Quantitative

médiane

FEATURE
ENGINEERING

- Métier
- Statistiques

ENCODAGE
STANDARDISATION

Qualitative

Label encoder
(binaire)
OHE

Quantitative

MinMaxScaler

MERGING

7 datasets ->
preprocessing +
feature
engineering -> 1
dataset

FEATURE
SELECTION

Corrélation
centre variables

Colonnes
inutiles

Feature
importances

Feature
selection:
- Boruta
- BoostAroota
- RFECV

*cf annexes : Erreurs, Valeurs manquantes, Imputation

PART II
FEATURE ENGINEERING
MERGING, FEATURE SELECTION

Step 1 : Feature Engineering (1/...)

VARIABLES QUANTITATIVES

AUTOMATIQUE

Création de variables
statistiques

['count', 'min', 'max', 'mean', 'var']

EXT_SOURCE_...

EXT_SOURCE_1
EXT_SOURCE_2
EXT_SOURCE_3

Données extérieures

sources moy, min, max, var
sources somme , somme pondérée

DAYS_-...

DAYS_BIRTH
DAYS_LAST_PHONE_CHANGE
DAYS_EMPLOYED
DAYS_ID_PUBLISH

Temps écoulé depuis...

diff âge - temps travaillé
ratio temps travaillé / âge

AMT_...

AMT_GOOD_PRICE

Crédit

ratio crédit / revenu
ratio revenu / annuité / âge
ratio crédit / annuité
ratio crédit / annuité / âge
crédit > demande ?
crédit > GoodPrice

FLOORMAX_...

FLOORMAX_AVG
FLOORMAX_MEDI

Renseignements domicile

domicile somme (moy, med, mode)

Famille

nombre d'adultes dans la famille
ratio revenu / nbre enfants
revenu par tête

Step 1 : Feature Engineering (2/...)

VARIABLES QUALITATIVES

MANUEL

REG_... / LIV_...

REG_CITY_NOT_WORK_CITY
REG_CITY_NOT_LIVE_CITY
REG_CITY_NOT_WORK_CITY

Lieu d'habitation
Lieu de travail
Ville
Région

flag région (sum)

Création de variables
« métier »

FLAG_DOC_... / FLAG_...

FLAG_DOCUMENT_3
FLAG_EMP_PHONE

Doc demandés
Contacts

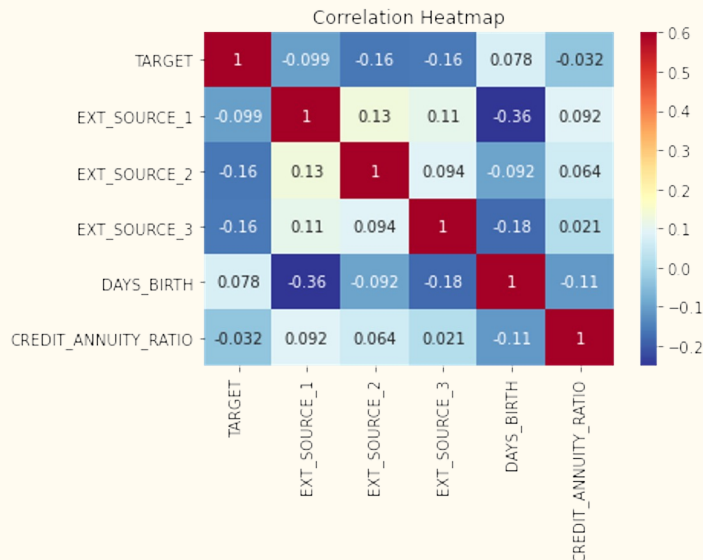
flag documents (sum)
flag contacts sum (phone, email, mobil)

Step 1 : Feature Engineering (3/...)



TARGET_NEIGHBORS_500_MEAN

Variable qui contient la moyenne de la variable 'TARGET' des 500 voisins d'une ligne particulière.
Les voisins sont calculés en utilisant les EXT_SOURCE et CREDIT_ANNUITY_RATIO.



Algorithme utilisé : **KNeighborsClassifier** de sklearn
k (nombre de voisins) = 500

1

Création de 2 dataframes : (à partir de train et test)
neighbors_train (307 511, 4) / neighbors_test (48 744, 4)
On récupère la cible 'TARGET' : train_target = train.TARGET

2

On entraîne le classificateur à l'aide de la méthode **fit()**
knn.fit(neighbors_train, train_target)

3

On récupère les **500 voisins** pour chaque ligne
train_500_neighbors = knn.kneighbors(neighbors_train)[1]
test_500_neighbors = knn.kneighbors(neighbors_test)[1]

4

On ajoute les moyennes de la cible des 500 voisins dans **une nouvelle colonne** :
train['TARGET_NEIGHBORS_500_MEAN'] = [train['TARGET'].iloc[ele].mean() for ele in train_500_neighbors]
test['TARGET_NEIGHBORS_500_MEAN'] = [train['TARGET'].iloc[ele].mean() for ele in test_500_neighbors]

Step 2 : Merging

Ordre d'assemblage des tables	Nombre de variables initial (par table)	Nombre de variables après FE
Application_train	122	209
bureau bureau_balance	17 3	51
previous_application	37	78
POS_CASH_balance	8	4
installments_payments	8	20
credit_card_balance	23	3
TOTAL	218	365

REMARQUES :

1. Une méthode de filtrage (corrélation de Pearson) est appliquée après FE à chaque table pour éliminer les **variables colinéaires**

2. **application_test** (48744 clients) : traité en parallèle pendant l'étude. contient les mêmes variables que le jeu utilisé pour l'entraînement du modèle. (Sauf variable cible) utilisé dans la partie **dashboard** pour simuler des nouveaux clients



train	(307511, 365)
test	(48744, 364)



**FEATURE
SÉLECTION**

Step 3 : Feature Selection

Jeu de donnée après feature engineering

(307511, 365)

La feature selection est un processus de sélection d'un sous-ensemble de variables qui sont les plus pertinentes pour la modélisation et l'objectif commercial du problème*

	Boruta	BoostAroota	LightGBM
Nombre de variables sélectionnées :	151	162	257
Taille jeu de données	(307511, 151)	(307511, 162)	(307511, 257)

Jeu de donnée après feature engineering

(307511, 109)



Variables communes

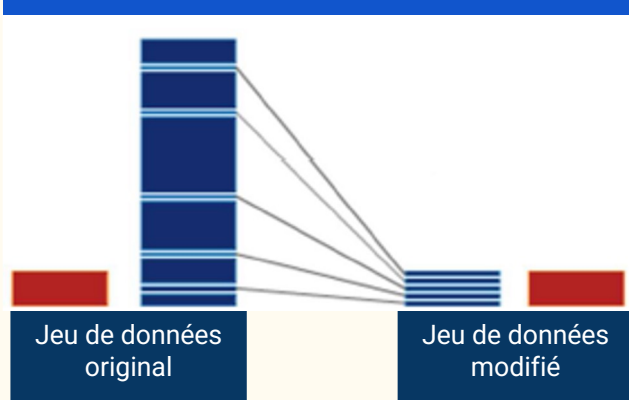
Tables après FS	Taille
train	(307511, 109)
test	(48744, 108)
1 ligne par client	

*cf annexes : Feature Selection

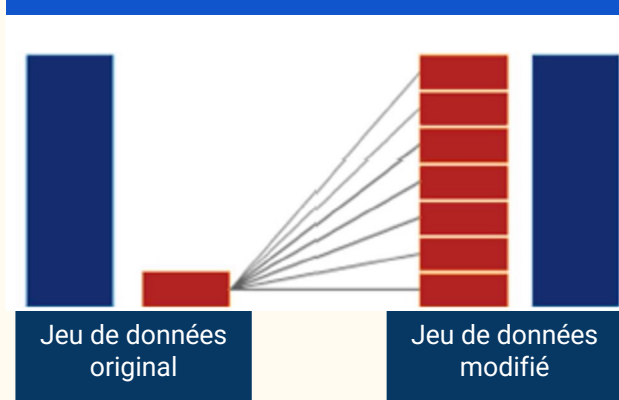
PART III
MODÉLISATION

Step 1 : Resampling (SMOTE / Modèle)

Undersampling : échantillonnage de la classe majoritaire

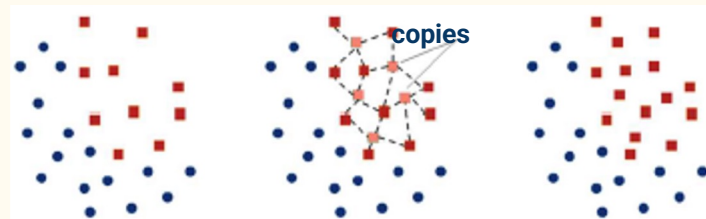


Oversampling : copies de la classe minoritaire

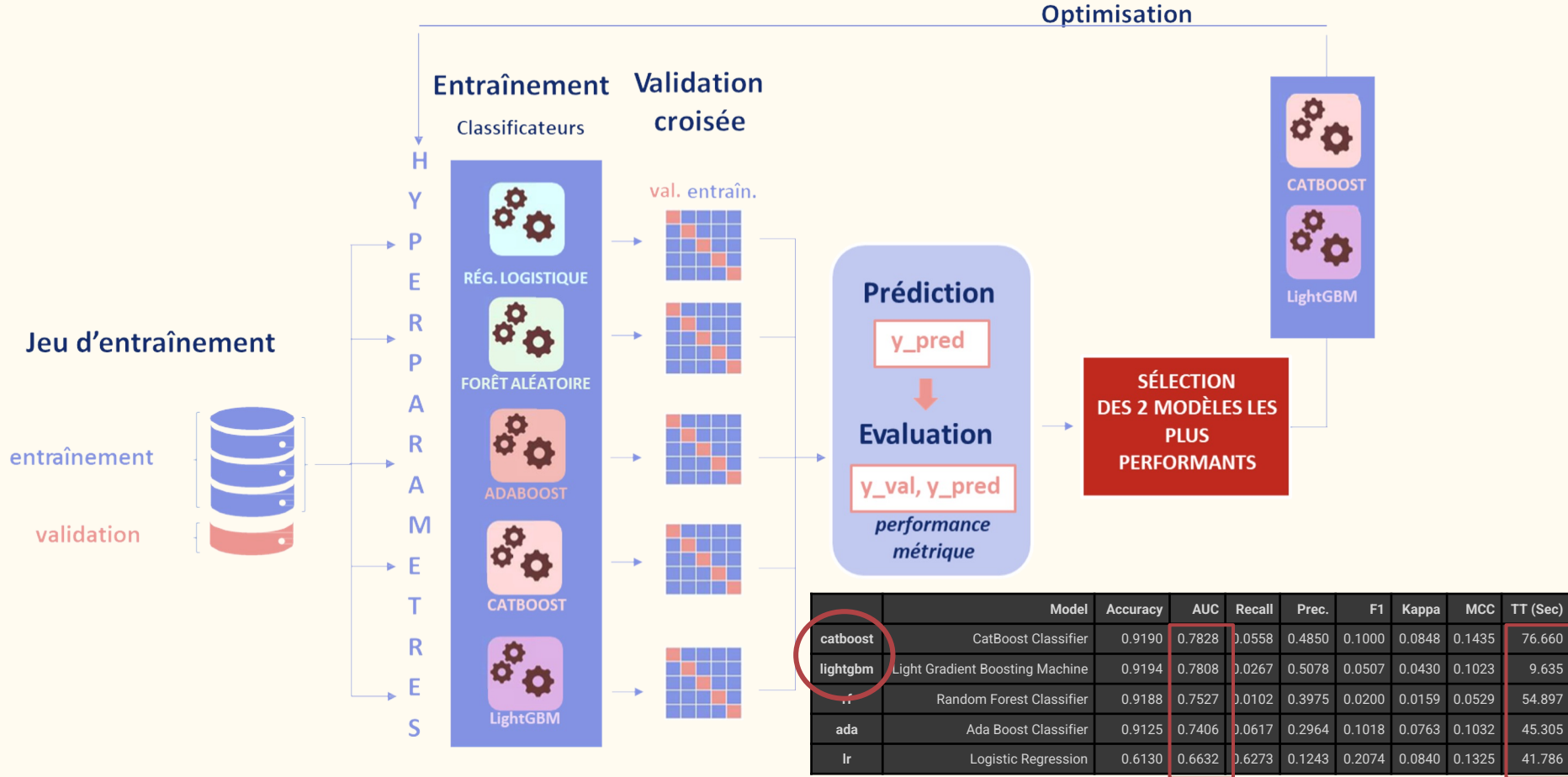


1 **SMOTE** (Synthetic Minority Oversampling Technique) Consiste à **synthétiser des éléments pour la classe minoritaire**, à partir de ceux qui existent déjà en choisissant aléatoirement un point de la classe minoritaire et à calculer les k plus proches voisins de ce point.

2 **Modèle** : on peut indiquer à certains modèles le déséquilibre en réglant un hyperparamètre (exemple : « **class_weight = 'balanced'** » pour LightGBM).



Step 2 : Model Selection avec Pycaret



Step 3 : Problématique métier, métrique et fonction de coût

Dans notre problème de classification binaire, le coût des **faux positifs** n'est pas le même que celui des **faux négatifs**.

Minimiser les pertes:

faux positifs : **non défaillants** prédits défaillants

-> l'organisme prêteur perd les intérêts que le prêt aurait générés.

faux négatifs : **défaillants** prédits non défaillants

-> l'organisme prêteur perd ainsi la somme prêtée.

Matrice de confusion

		Classe prédite	
		Classe 0 : non-défaillant	Classe 1 : défaillant
Classe réelle	Classe 0 : non-défaillant	Vrais négatifs TN	Faux Positifs FP
	Classe 1 : défaillant	Faux Négatifs FN	Vrais Positifs

Un intérêt plus grand sera porté aux **faux négatifs**, encore plus coûteux que les **faux positifs**



CRÉATION D'UNE MÉTRIQUE ET D'UNE FONCTION COÛT

Choix arbitraire de pénalisation :

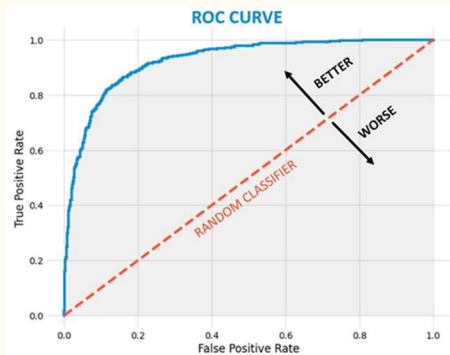
- Mauvais prêts : pénalisation de -10
- Bons prêts : gain de 1

Step 4 : Optimisation du meilleur modèle (1/2)

LightGBM, plus rapide, retenu. **Optimisation** : selon 2 métriques et sur 2 jeux de données

1

Métrique : **Score AUC** (aire sous la courbe). Plus le modèle est performant, plus l'aire sous la courbe est maximisé.



MÉTRIQUES

1

Jeu de données **rééquilibré**
avec **SMOTE**
(OVERSAMPLING)

JEUX DE
DONNÉES

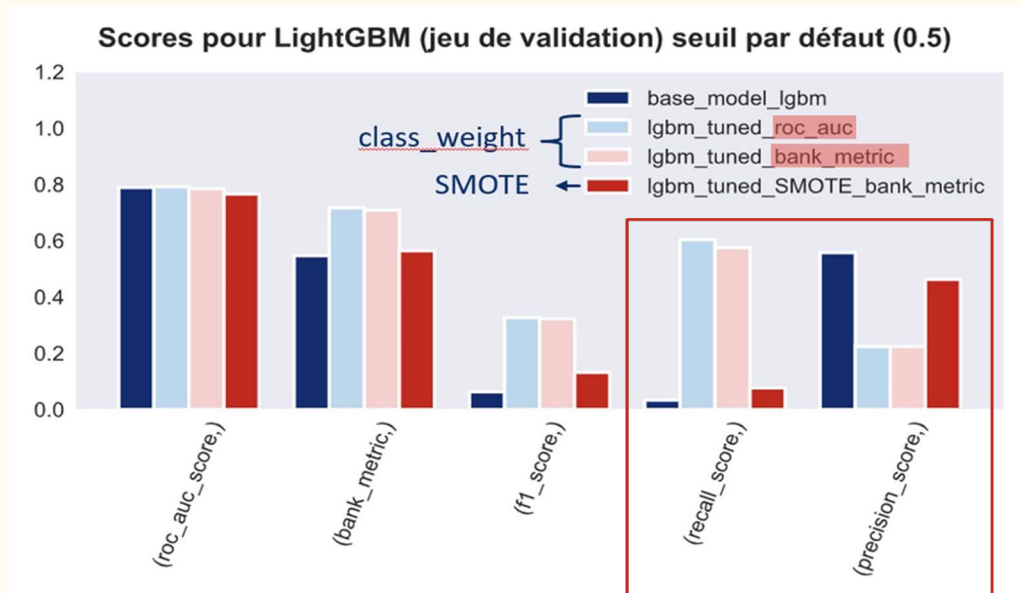
2

Jeu de données **non** rééquilibré
Réglage LightGBM
: « **class_weight =**
'balanced' »

2

Métrique : « **bancaire** » créée par nous mêmes, permettant de pénaliser les erreurs les plus coûteuses et donc limiter les pertes.

Step 4 : Optimisation du meilleur modèle (2/2)



1. Rappel et précision

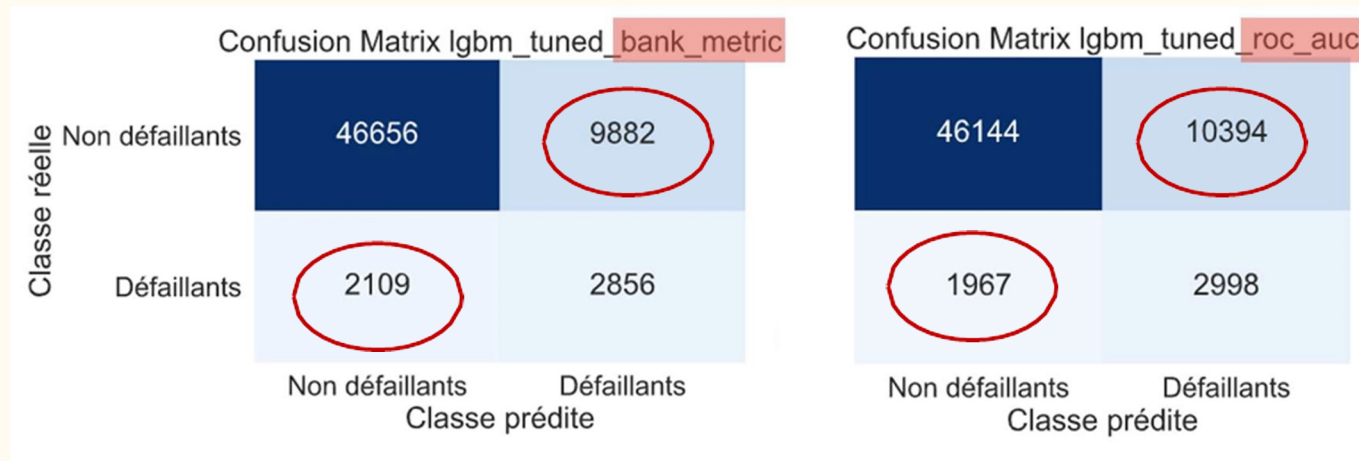
But dans le cas d'une classification de prêts bons ou mauvais il faut :

Maximiser le rappel au détriment de la précision (diminuer les faux négatifs pour augmenter le rappel)

2. Conclusion LightGBM

L'hyperparamètre `class_weight = 'balanced'` donne de meilleurs résultats. C'est la **stratégie de rééquilibrage** que nous choisirons.

Step 5 : Meilleur modèle et seuil de solvabilité (1/2)



faux négatifs : LightGBM « **métrique bancaire** » < LightGBM **ROC_AUC** -> **perte de la somme prêtée**

faux positifs : LightGBM « **métrique bancaire** » > LightGBM **ROC_AUC** -> **perte des intérêts**



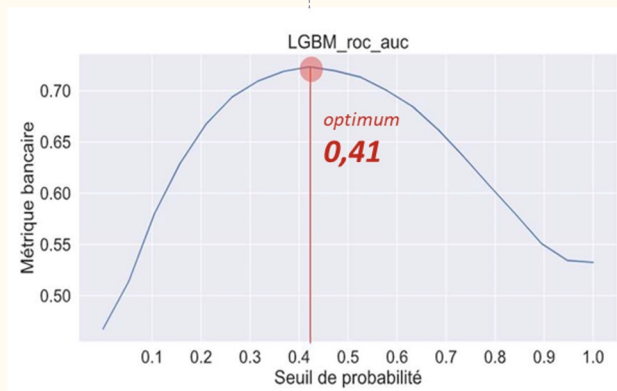
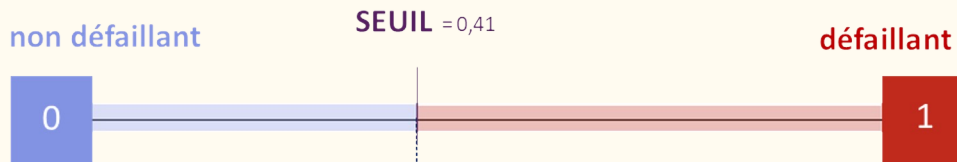
MODÈLE CONSERVÉ :
LightGBM roc_auc

Métrique bancaire utilisée pour fixer le **seuil de solvabilité**

Step 5 : Meilleur modèle et seuil de solvabilité (2/2)

MODÈLE CONSERVÉ :
LightGBM roc_auc

Métrique bancaire
utilisée pour fixer le
seuil de solvabilité

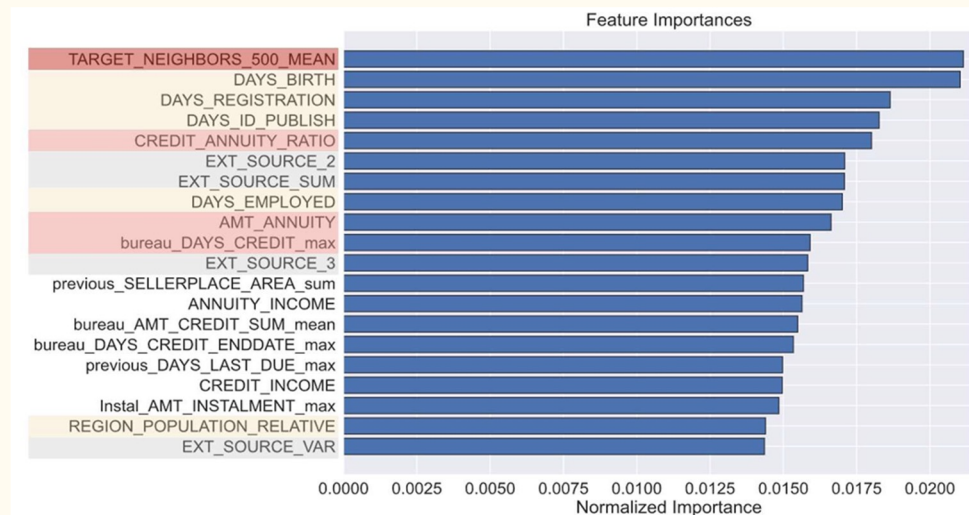


Confusion Matrix lgbm_tuned_roc_auc

46144	10394
1967	2998
Non défaillants	Défaillants
Classe prédite	

Step 6 : Modèle optimisé et interprétabilité LGBMClassifier

	Défaut	Optimisé
n_estimators	100	10 000
learning_rate	0,1	0,05
objective	None	'binary'
class_weight	None	'balanced'
boosting_type	'gbdt'	'gbdt'
num_leaves	31	48
max_depth	-1	11
min_split_gain	0	0,1
min_child_weight	0,001	80
min_child_samples	20	18
subsample	1	0,73
colsample_bytree	1	0,67
reg_alpha	0	0,3
gamma	0	0,15



lgbm_tuned_roc_auc

roc_auc_score	0.791848
bank_metric	0.716879
f1_score	0.326633
recall_score	0.603827
precision_score	0.223865

- Variables bancaires
- Variables personnelles
- Variables externes

*cf annexe : Feature Importance

PART IV

DASHBOARD

SIDEBAR

INFORMATIONS CLIENTS

âge, sexe, situation familiale, ancienneté, revenu

INFOS GRAPHIQUES

Infos graphiques et statistiques supplémentaires tirés de l'analyse exploratoire de données.

Stats : Income type

Income type of the selected client : Working

Distribution of income type



Percentage of defaulters for each category of income type



Client informations

Age 31 years

Gender : Female

Family status : Married

Education : Higher education

Years employed 3 years

Income type : Working

Income 450000 \$

Contract type : Cash loans

More informations

Stats and client infos

Age

Gender

Family status

Education

Years employed

Income type

Income

Contract type

SHAP explainer

Explain Results by SHAP

PRET A DEPENSER

"interactif dashboard"

1

Client ID

Please select a client ID

208550

	SK_ID_CURR	Prediction	Prediction_Score	GENDER	YEARS_BIRTH	YE
0	208550	0	3.079450257647151	Female	31.09041095890411	3.147

2

Default Risk Score

Select the threshold : (default : 0.41)

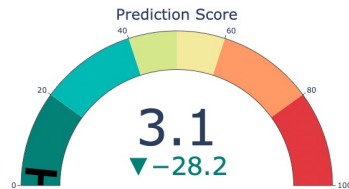
Threshold:

0.00 1.00

a

Prediction of the selected client with the current threshold : Non defaulter

b



TRUST score for the selected client : EXCELLENT

Prediction Score for similar clients : 31.3

APPLICATION PRINCIPALE

1- Sélection d'un client

2- Risque de défaut de paiement

a) Réglage du seuil de solvabilité
Seuil réglable de 0 à 1 (défaut : 0,41)

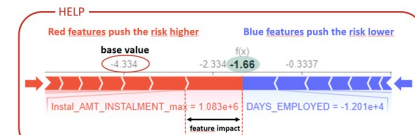
b) Jauge de prédiction

- Score de prédiction de 0 à 100 associé à un qualificatif
- Comparaison avec le score des 20 plus proches voisins

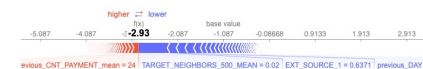
SHAP : explain results

How most important features impacts Class prediction?
Force plot shows, how opposite are the features strengths

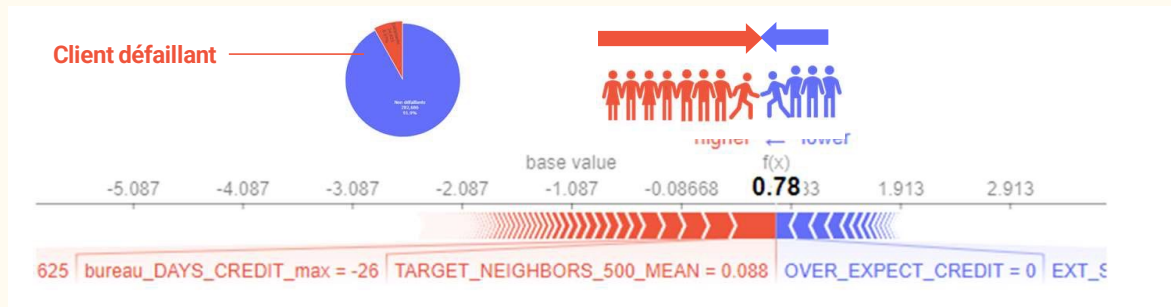
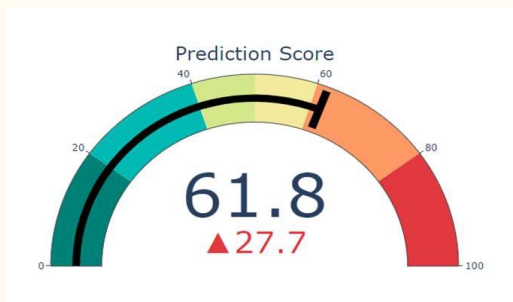
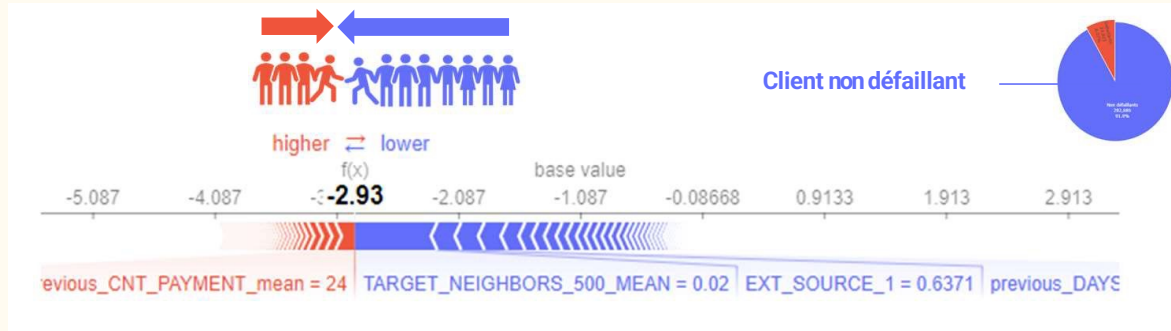
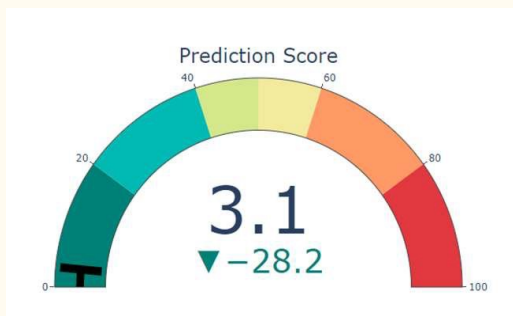
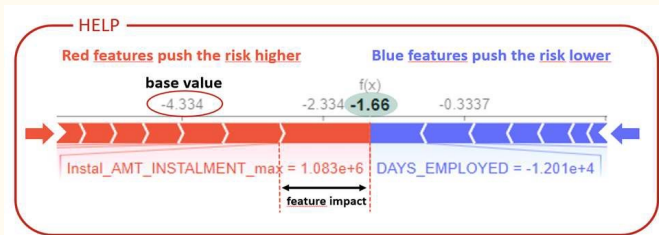
SHAP HELP



SHAP Force plot for the selected client



SHAP

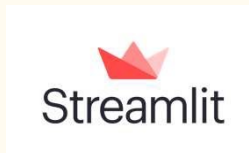


Deploiement du dashboard



Dépôt GitHub :

https://github.com/EricBlanvillain/P7_blanvillain_eric/tree/main/dashboard



En local :

```
cd dashboard  
streamlit run dashboard.py
```

A distance :

https://share.streamlit.io/ericblanvillain/p7_blanvillain_eric/main/dashboard/dashboard.py

CONCLUSION

Conclusions et axes d'améliorations

CONCLUSIONS

Notre étude portait sur un problème de **classification binaire présentant un déséquilibre de classe**.

Modèle final : **LightGBM** optimisé sur la métrique ROC_AUC.

Mise en place de stratégies pour optimiser le meilleur modèle et obtenir une performance maximale:

1. **différentes solutions de rééquilibrage de classe** testées et comparées
2. **création de nouvelles variables** facilement explicables (demande client)
3. **création d'une métrique métier** et fixation d'un **seuil de solvabilité** optimum.

AMÉLIORATIONS POSSIBLES

1. **Optimisation** plus fine des hyperparamètres du modèle
2. **Modification** de la métrique créée, avec l'aide d'un expert métier
3. **Création de variables** plus pertinentes avec l'expert

Bibliographie

- “Home Credit Default Risk.” Kaggle (1st Place Solution - Discussion). <https://www.kaggle.com/c/home-credit-default-risk/discussion/64821>
- Rao, Rishabh. “Home Credit Default Risk-an End to End ML Case Study-Part 1: Introduction and Eda.” Medium. TheCyPhy, November 1, 2020. <https://medium.com/thecyphy/home-credit-default-risk-part-1-3bfe3c7ddd7a>
- Rao, Rishabh. “Home Credit Default Risk-an End to End ML Case Study-Part 2: Feature Engineering and Modelling.” Medium. TheCyPhy, November 1, 2020. <https://medium.com/thecyphy/home-credit-default-risk-part-2-84b58c1ab9d5>
- Narkhede, Sarang. “Understanding AUC - Roc Curve.” Medium. Towards Data Science, June 15, 2021. <https://towardsdatascience.com/understanding-auc-roc-curve-68b2303cc9c5>
- Residentmario. “Automated Feature Selection with Boruta.” Kaggle. Kaggle, April 20, 2018. <https://www.kaggle.com/residentmario/automated-feature-selection-with-boruta>
- Dansbecker. “Shap Values.” Kaggle. Kaggle, November 9, 2021. <https://www.kaggle.com/dansbecker/shap-values>
- Charfaoui, Younes. “Hands-on with Feature Selection Techniques: Embedded Methods.” Medium. Heartbeat, September 24, 2021. <https://heartbeat.comet.ml/hands-on-with-feature-selection-techniques-embedded-methods-84747e814dab>

ANNEXES

Erreurs / Aberrations détectées

Alignement des 2 jeux de données

- feature CODE_GENDER has different values: {'XNA'}
- feature NAME_INCOME_TYPE has different values: {'Maternity leave'}
- feature NAME_FAMILY_STATUS has different values: {'Unknown'}

Variable 'CODE_GENDER'

Le jeu d'entraînement contient seulement 4 valeurs nommés 'XNA' pour la colonne renseignant le genre

Variable 'NAME_INCOME_TYPE'

La colonne 'NAME_INCOME_TYPE' prend la valeur 'Maternity leave' uniquement pour le jeu d'entraînement , et pour seulement 5 emprunteurs

Variable 'NAME_FAMILY_STATUS' De la même manière, pour la colonne NAME_FAMILY_STATUS, il y a seulement deux fois la valeur Unknown et uniquement pour le jeu d'entraînement.

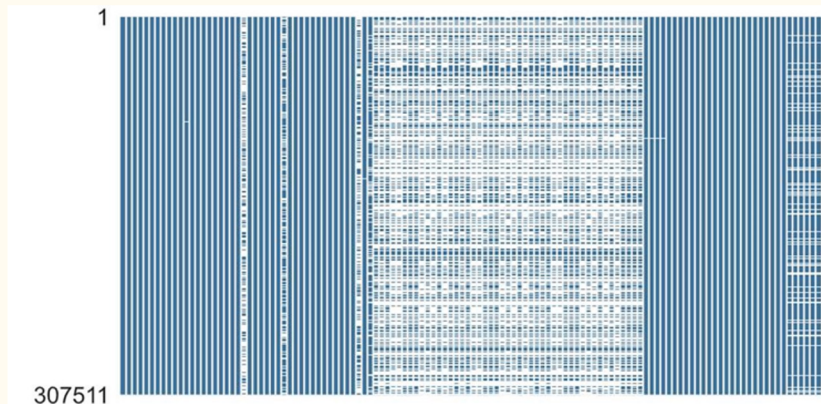
Correction des aberrations détectées lors de l'EDA

Suppression des aberrations détectées chez la variable **"DAYS EMPLOYED "**

Suppression des aberrations détectées chez les variables **"OBS"**

Valeurs manquantes

	Nombre de valeurs manquantes	% de valeurs manquantes
COMMONAREA_MEDI	214865	69.870000
COMMONAREA_MODE	214865	69.870000
COMMONAREA_AVG	214865	69.870000
NONLIVINGAPARTMENTS_MODE	213514	69.430000
NONLIVINGAPARTMENTS_MEDI	213514	69.430000
NONLIVINGAPARTMENTS_AVG	213514	69.430000
FONDKAPREMONT_MODE	210295	68.390000
LIVINGAPARTMENTS_MEDI	210199	68.350000
LIVINGAPARTMENTS_AVG	210199	68.350000
LIVINGAPARTMENTS_MODE	210199	68.350000
FLOORSMIN_MODE	208642	67.850000
FLOORSMIN_AVG	208642	67.850000
FLOORSMIN_MEDI	208642	67.850000
YEARS_BUILD_MEDI	204488	66.500000
YEARS_BUILD_MODE	204488	66.500000
YEARS_BUILD_AVG	204488	66.500000
OWN_CAR_AGE	202929	65.990000
LANDAREA_AVG	182590	59.380000
LANDAREA_MODE	182590	59.380000
LANDAREA_MEDI	182590	59.380000
BASEMENTAREA_MEDI	179943	58.520000
BASEMENTAREA_MODE	179943	58.520000
BASEMENTAREA_AVG	179943	58.520000
EXT_SOURCE_1	173378	56.380000
NONLIVINGAREA_MODE	169682	55.180000
NONLIVINGAREA_AVG	169682	55.180000
NONLIVINGAREA_MEDI	169682	55.180000
ELEVATORS_MODE	163891	53.300000
ELEVATORS_AVG	163891	53.300000
ELEVATORS_MEDI	163891	53.300000

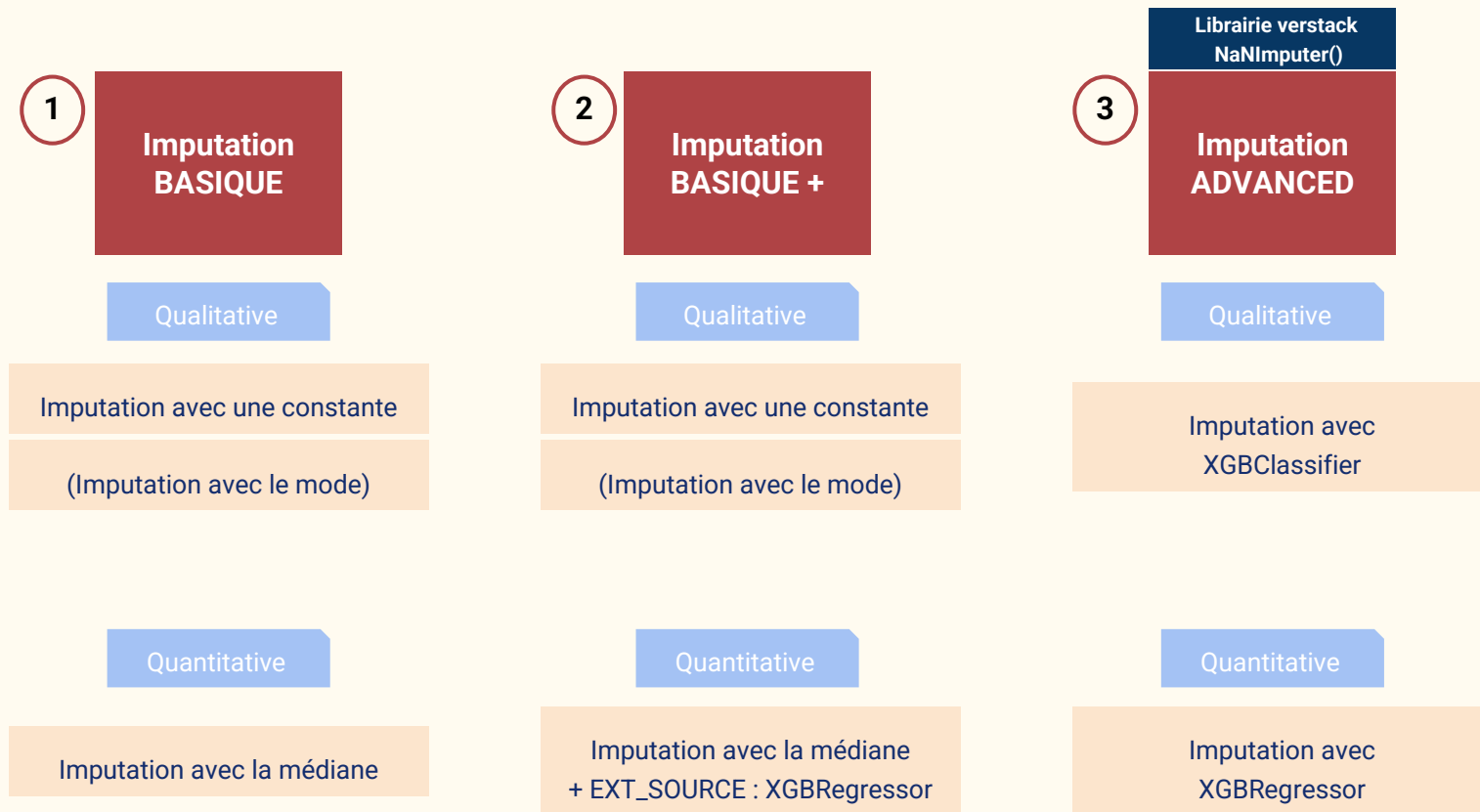


Suppression des colonnes NAN > 60%

Variables supprimées :

[OWN_CAR_AGE, YEARS_BUILD_AVG, COMMONAREA_AVG, FLOORSMIN_AVG, LIVINGAPARTMENTS_AVG, NONLIVINGAPARTMENTS_AVG, YEARS_BUILD_MODE, COMMONAREA_MODE, FLOORSMIN_MODE, LIVINGAPARTMENTS_MODE, NONLIVINGAPARTMENTS_MODE, YEARS_BUILD_MEDI, COMMONAREA_MEDI, FLOORSMIN_MEDI, LIVINGAPARTMENTS_MEDI, NONLIVINGAPARTMENTS_MEDI, FONDKAPREMONT_MODE]

Imputation : trois façons de procéder



Feature Selection

Méthodes de Feature Selection

La feature selection est un processus de sélection d'un sous-ensemble de variables qui sont les plus pertinentes pour la modélisation et l'objectif commercial du problème

FILTRAGE

Corrélation de Pearson

- Chi 2
- F Test
- ANOVA
- Information Gain

WRAPPER

Boruta

BoostAroota

Heuristics :

- Forward Selection
- Backward Elimination
- Recursive Feature Elimination

Methodical :

- Best First Search
- DFS

Stochastic :

- Random Hill
- Climbing
- Simulated Annealing

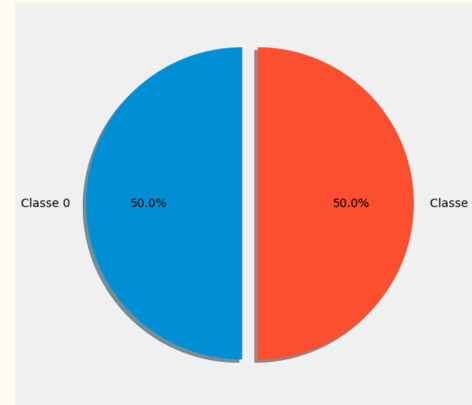
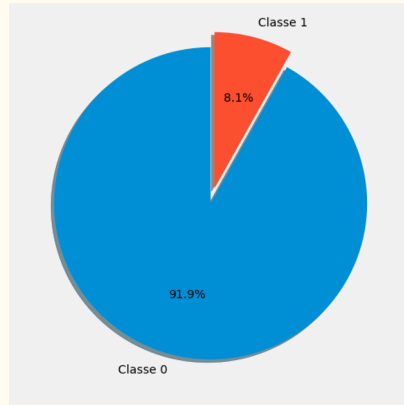
EMBEDDED

LightGBM

- Lasso Regression
- Ridge Regression
- Elastic Nets
- Decision Trees
- RF

Resampling (SMOTE)

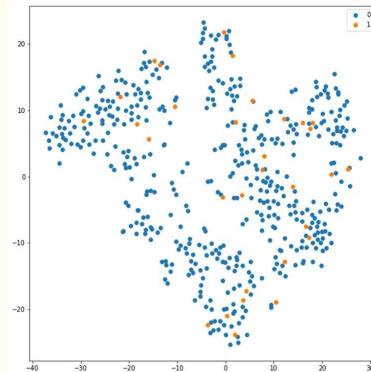
1 Defaillants
0 Non défaillants



Initial

Échantillon : 500

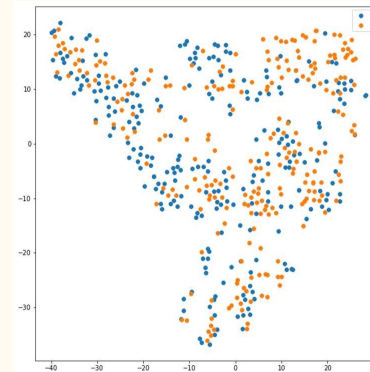
1 0.10
0 0.90



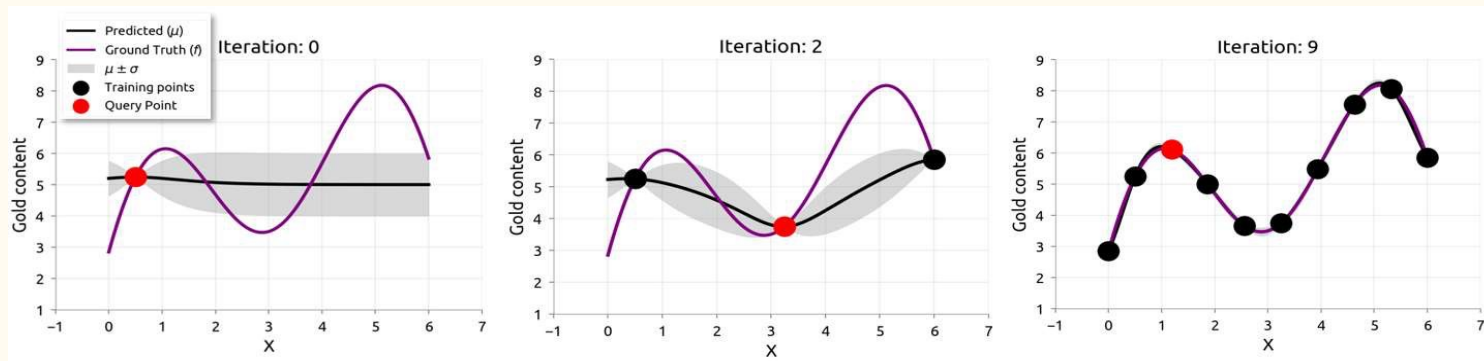
SMOTE

Échantillon : 500

1 0.45
0 0.55



Optimisation bayésienne du modèle LightGBM



+ Trace gardée des résultats d'évaluation passés
pour former un modèle probabiliste

L'optimisation bayésienne construit un modèle de probabilité de la fonction objectif afin de **proposer des choix plus intelligents pour le prochain ensemble d'hyperparamètres à évaluer**. Au fur et à mesure que le nombre d'observations augmente, la distribution postérieure s'améliore et l'algorithme devient plus sûr des régions de l'espace des paramètres qui valent la peine d'être explorées et de celles qui ne le sont pas.

Optimisation des paramètres du modèle LightGBM

Hyperparamètres	Descriptions	Notes	Nos hyperparamètres
num_estimators	Le nombre maximum d'arbres qui peuvent être construits lors de la résolution de problèmes d'apprentissage automatique.	Utiliser un très grand nombre d'itérations si utilisation de l'arrêt précoce.	num_estimators: 10 000
learning_rate	Le taux d'apprentissage.	En général, nous utilisons un taux d'apprentissage de 0,05 ou moins pour la formation, tandis qu'un taux d'apprentissage de 0,10 ou plus est utilisé pour modifier les hyperparamètres.	learning_rate: 0.05
max_depth	Profondeur de l'arbre.	Une valeur plus grande est généralement meilleure, mais la vitesse d'overfitting augmente. Typique : 6, généralement [3, 12].	max_depth: 11
lamda_l1 lambda_l2	Régularisation L1 pour le boosting Régularisation L2 pour le boosting		reg_alpha: 0,3 reg_lambda: 0,15
colsample_bytree	Rapport de sous-échantillonnage des colonnes lors de la construction de chaque arbre.		colsample_bytree: 0,67
subsample	Rapport de sous-échantillon de l'instance d'apprentissage.		subsample: 0,73
num_leaves	Le nombre maximum de feuilles dans l'arbre résultant.		num_leaves: 48
min_split_gain	Réduction de la perte minimale requise pour effectuer une partition supplémentaire sur un nœud feuille de l'arbre. Le nombre minimum d'échantillons d'entraînement dans une feuille.	Cette technique est extrêmement utile lorsque vous essayez de construire des arbres profonds, mais que vous essayez également d'éviter de construire des branches inutiles de ces arbres (overfitting).	min_split_gain: 0,1
min_child_weight	Somme minimale de poids d'instance (hessian) nécessaire dans un enfant (feuille).		min_child_weight: 80
min_child_samples	Nombre minimum de données nécessaires dans un enfant (feuille).		min_child_samples: 18
objective	Binary Description : Application sigmoïde comme fonction d'activation. Entropie croisée comme fonction de perte.		binary

Feature importance : LightGBM

TARGET_NEIGHBOORS_500			Valeur TARGET moyenne des 500 voisins les plus proches de chaque ligne, où chaque voisinage est défini par les trois sources externes (1,2,3) et le ratio crédit/intérêts
DAYS_BIRTH			Âge du client en jours au moment de la demande
DAYS_REGISTRATION			Combien de jours avant la demande le client a-t-il modifié son inscription ?
DAYS_ID_PUBLISH			Combien de jours avant la demande le client a-t-il modifié le document d'identité avec lequel il avait demandé le prêt?
CREDIT_ANNUITY_RATIO			Ratio montant du crédit/annuité
EXT_SOURCE_2			Score normalisé provenant d'une source de données externe
EXT_SOURCE_SUM			Somme des 3 variables EXT_SOURCE
DAYS_EMPLOYED			Combien de jours avant la demande la personne a-t-elle commencé son emploi actuel
AMT_ANNUITY			Intérêts du prêt
DAYS_CREDIT_max			Combien de jours avant la demande actuelle le client a-t-il fait une demande de crédit auprès de Home Credit
EXT_SOURCE_3			Score normalisé provenant d'une source de données externe
SELLERPLACE_AREA_sum			Zone de vente du lieu de vente de la demande précédente
ANNUITY_INCOME			Ratio intérêt/revenu du client
AMT_CREDIT_SUM_mean			Score normalisé provenant d'une source de données externe

Variables bancaires



Variables personnelles




Variables externes




Feature engineering



 app_train

 bureau

 previous