Formation Data Scientist

Soutenance Projet 7 :
Implémentez un modèle de scoring

-- Mélanie WARY --

Decembre 2021

PROBLÉMATIQUE

Mise en œuvre d'un **outil de "scoring crédit"** pour calculer la probabilité qu'un client rembourse son crédit, puis **classifier la demande** en crédit accordé ou refusé.

MISSIONS

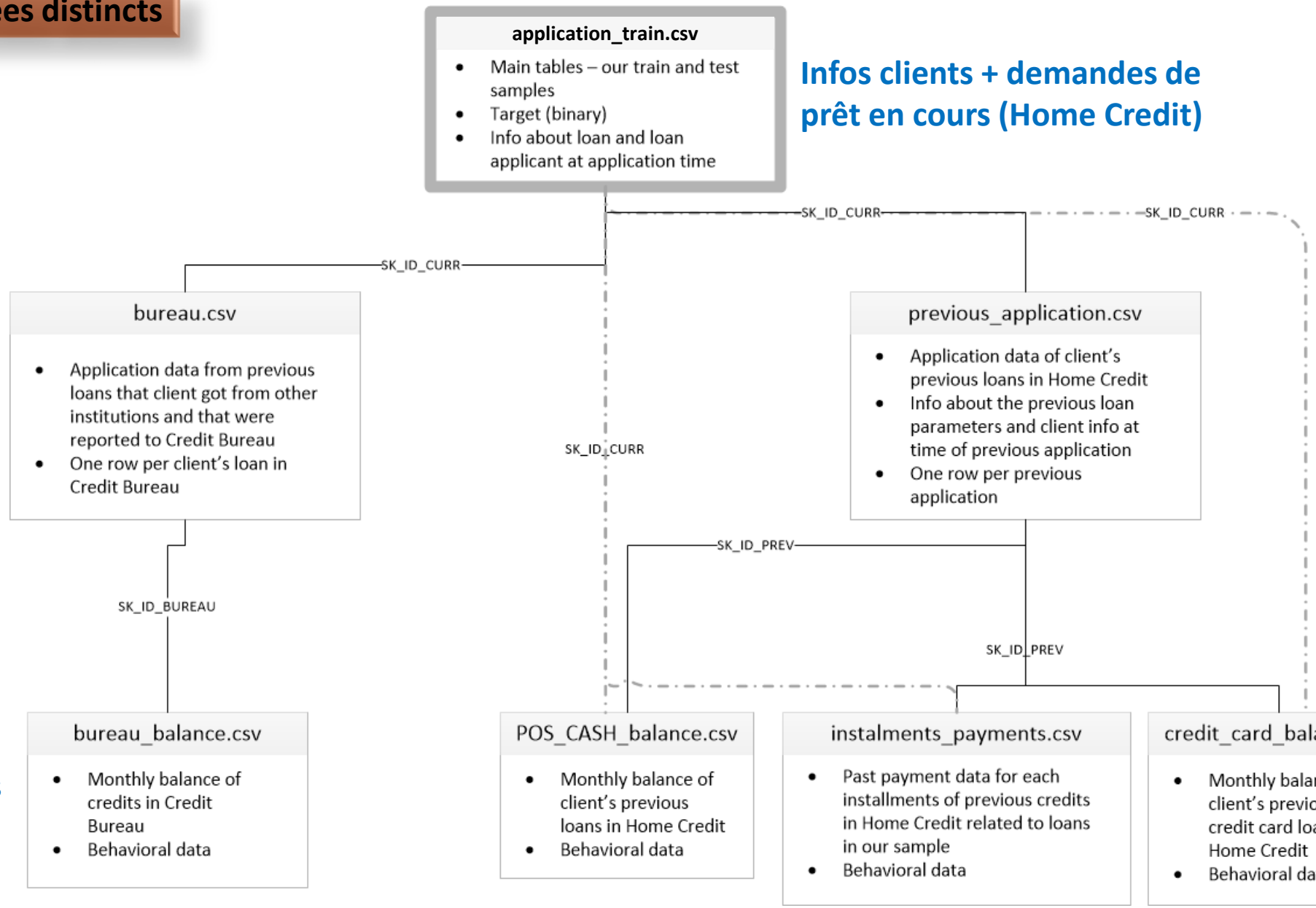
❶ Construire un **modèle** de scoring donnant une prédiction sur la probabilité de faillite d'un client de façon automatique.

❷ Construire un **dashboard** interactif à destination des gestionnaires de la relation client permettant d'interpréter les prédictions faites par le modèle

INTERPRÉTATION :

- **Nettoyage, analyse et feature engineering** des données à disposition à partir de **kernels Kaggle** à adapter.
- **Tests de différents modèles de classification supervisés** et **méthodes** permettant de pallier aux **déséquilibres des classes à prédire**.
- **Optimisation** du **modèle** et de la **méthode** les plus performants + optimisation du **seuil de probabilité** conditionnant l'attribution des classes.
- **Déploiement du modèle** sous forme d'**API**
- **Elaboration et déploiement** du **dashboard** permettant d'interpréter les prédictions du modèle.

7 jeux de données distincts



Infos clients + demandes de prêt en cours (Home Credit)

Anciennes demandes de prêt auprès d'autres banques

+

données comportementales clients sur ces anciens prêts

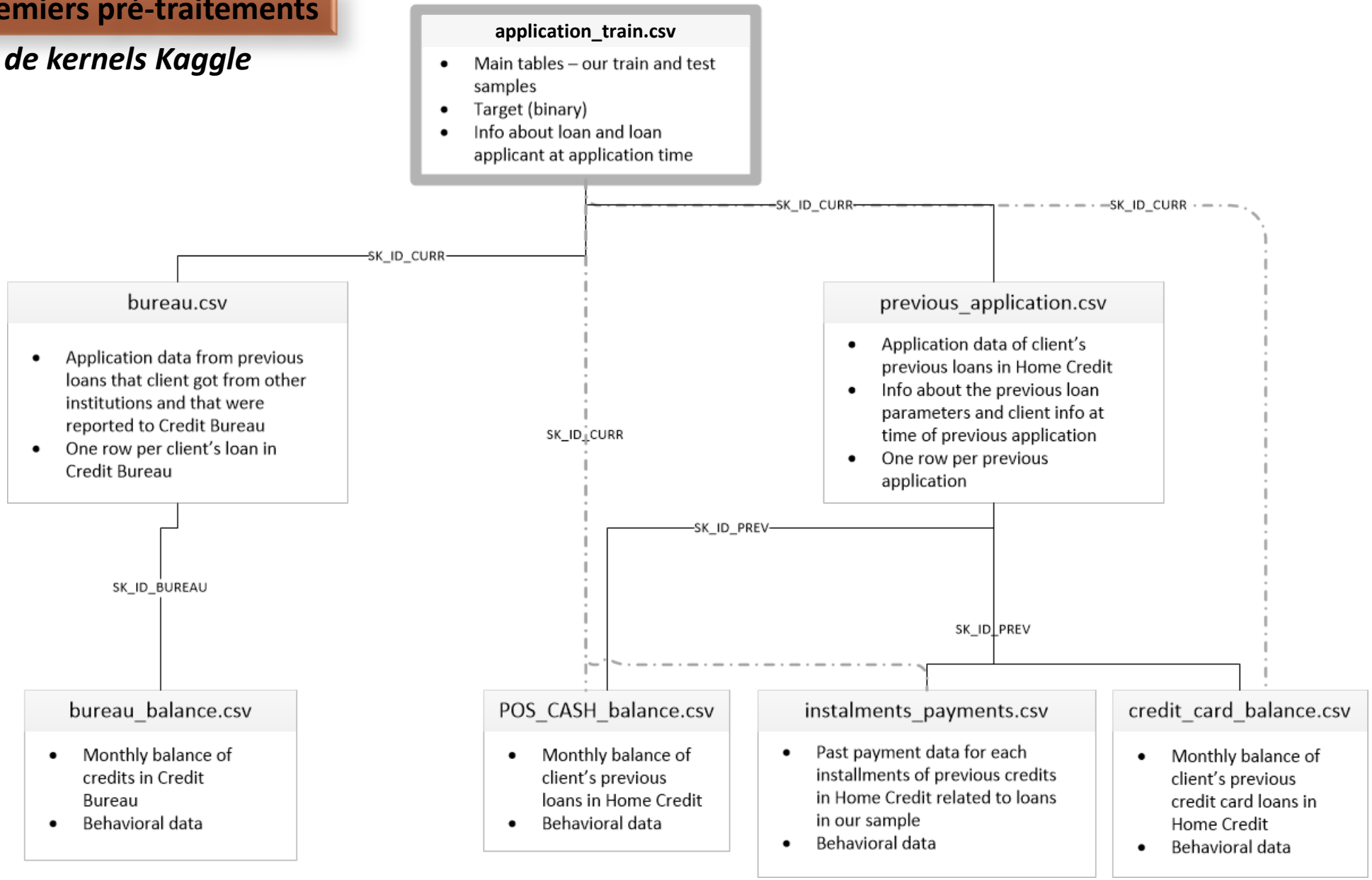
Anciennes demandes de prêt auprès de Home Credit

+

Données comportementales

Nettoyage et premiers pré-traitements

> Adaptation de kernels Kaggle

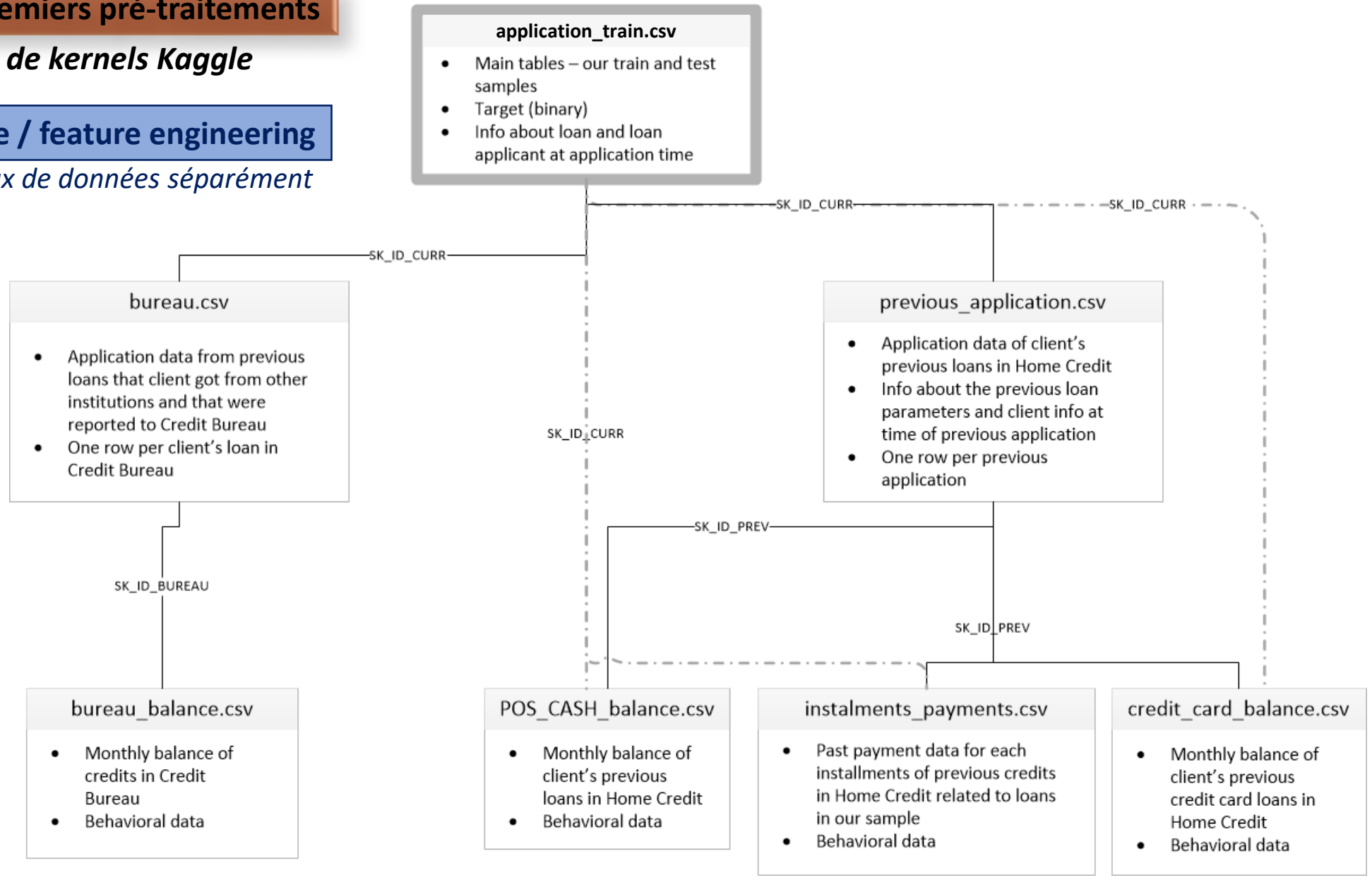


Nettoyage et premiers pré-traitements

> Adaptation de kernels Kaggle

1 **Nettoyage / feature engineering**

sur les 7 jeux de données séparément



Nettoyage et premiers pré-traitements

> Adaptation de kernels Kaggle

1 **Nettoyage / feature engineering**

sur les 7 jeux de données séparément

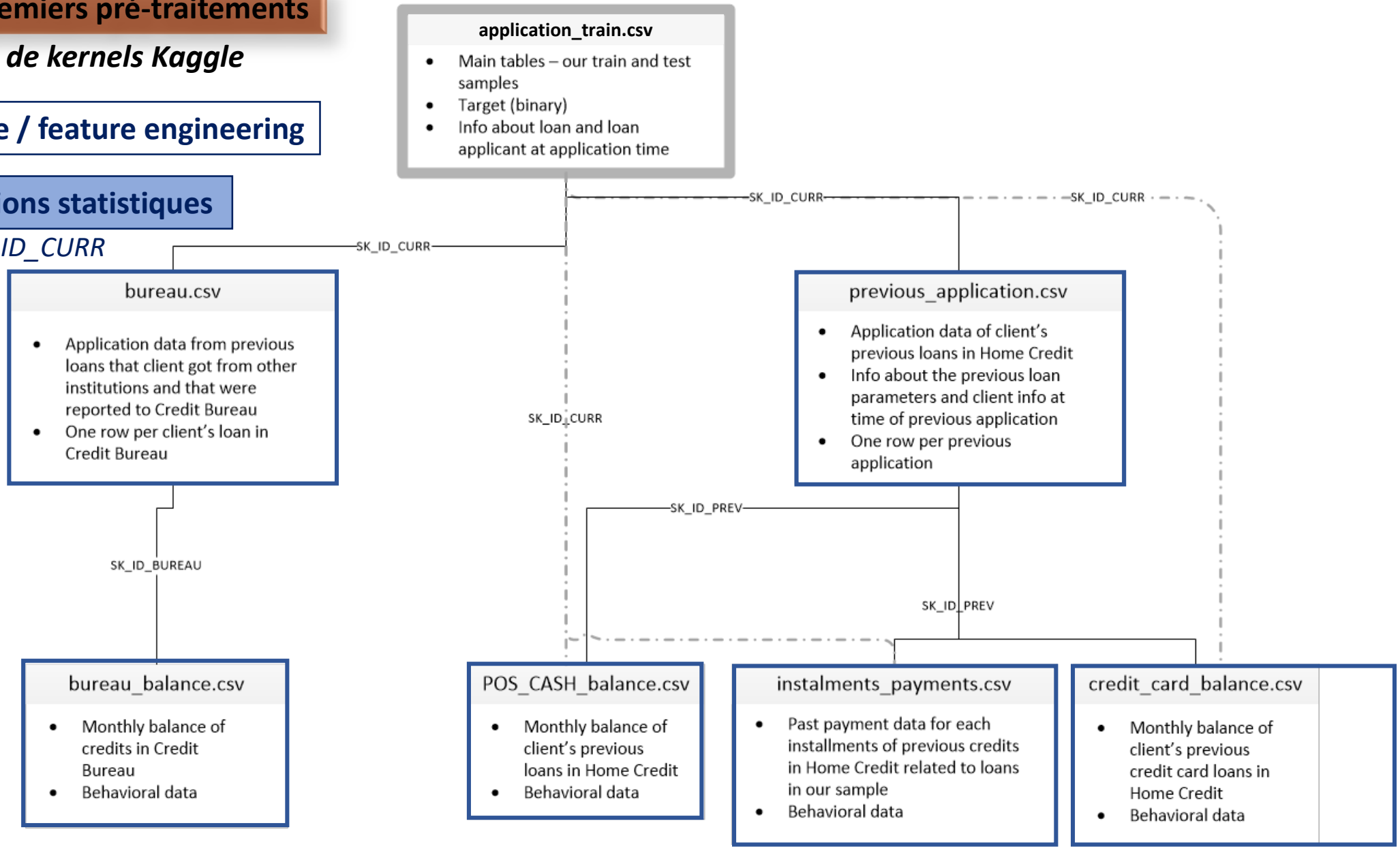
- Traiter les **valeurs rares** de variables catégorielles
- Traiter les **valeurs aberrantes**,
- appliquer une **transformation sinus et cosinus aux variables cycliques**,
- créer de **nouvelles variables** sur la base de celles fournies qui soient potentiellement plus informatives pour le modèle,
- **encoder les variables catégorielles** – non supportées par certains modèles.

Nettoyage et premiers pré-traitements

> Adaptation de kernels Kaggle

1. **Nettoyage / feature engineering**
2. **Aggrégations statistiques**

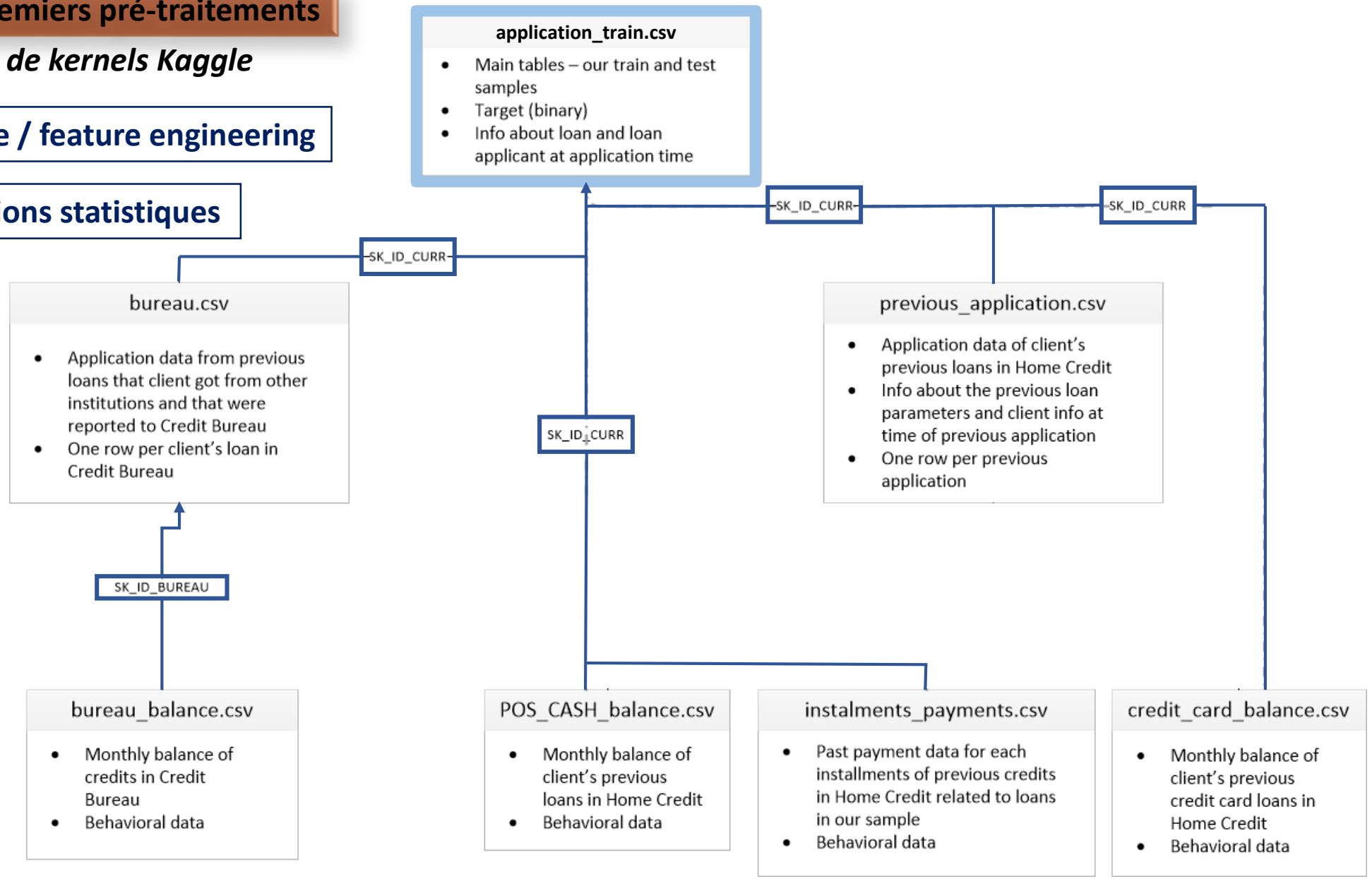
sur SK_ID_CURR



Nettoyage et premiers pré-traitements

> Adaptation de kernels Kaggle

1. **Nettoyage / feature engineering**
2. **Aggrégations statistiques**
3. **Jointure**



Nettoyage et premiers pré-traitements

> Adaptation de kernels Kaggle

1. **Nettoyage / feature engineering**
2. **Aggrégations statistiques**
3. **Jointure** ----------> Jeu de données à **778 variables** (et 307507 demandes de prêt)

Nettoyage et premiers pré-traitements

> Adaptation de kernels Kaggle

1. **Nettoyage / feature engineering**
2. **Aggrégations statistiques**
3. **Jointure** ----------> Jeu de données à **778 variables** (et 307507 demandes de prêt)
4. **Réduction du nombre de variables**

Suppression :

- des variables avec plus de **75% de données manquantes**
- des variables présentant la **même valeur** sur l'ensemble des individus (encodage)
- d'une des deux variables de chaque paire de **variables fortement corrélées** (coefficient de corrélation > 0.9 ou <-0.9)
  - ➥ données manquantes de la variable conservée comblées par les données de la variable supprimée.
- des variables montrant **peu de différences significatives entre les classes à prédire** (ANOVA f-score < 200).

Nettoyage et premiers pré-traitements

> Adaptation de kernels Kaggle

1. **Nettoyage / feature engineering**
2. **Aggrégations statistiques**
3. **Jointure** ----------> Jeu de données à **778 variables** (et 307507 demandes de prêt)

⬇

4. **Réduction du nombre de variables** ----------> Jeu de données à **129 variables** (et 307507 demandes de prêt)

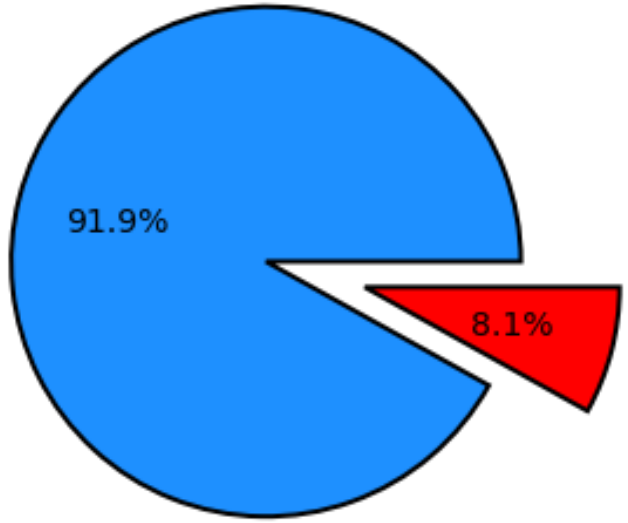
Nettoyage et premiers pré-traitements

> Adaptation de kernels Kaggle

1. **Nettoyage / feature engineering**
2. **Aggrégations statistiques**
3. **Jointure**
4. **Réduction du nombre de variables** ----------> Jeu de données à **129 variables** (et 307507 demandes de prêt)
5. **Partage stratifié en jeu d'entrainement et jeu de test**
   70% *30%*

Distribution non homogène des classes à prédire

Classe '0' : (classe majoritaire / négative) -- Crédit accordé / remboursé



Classe '1' : (classe minoritaire / positive) -- Crédit non accordé / non remboursé

Pipeline de base

Données nettoyées, pré-traitées et partagées en training et testing sets

PIPELINE

Imputation NaN par la médiane

Standardisation des variables numériques non binaires

Sur/sous-échantillonnage des classes minoritaires/majoritaires

Classifieur binaire

Test et sélection des méthodes de pré-traitements et modèle

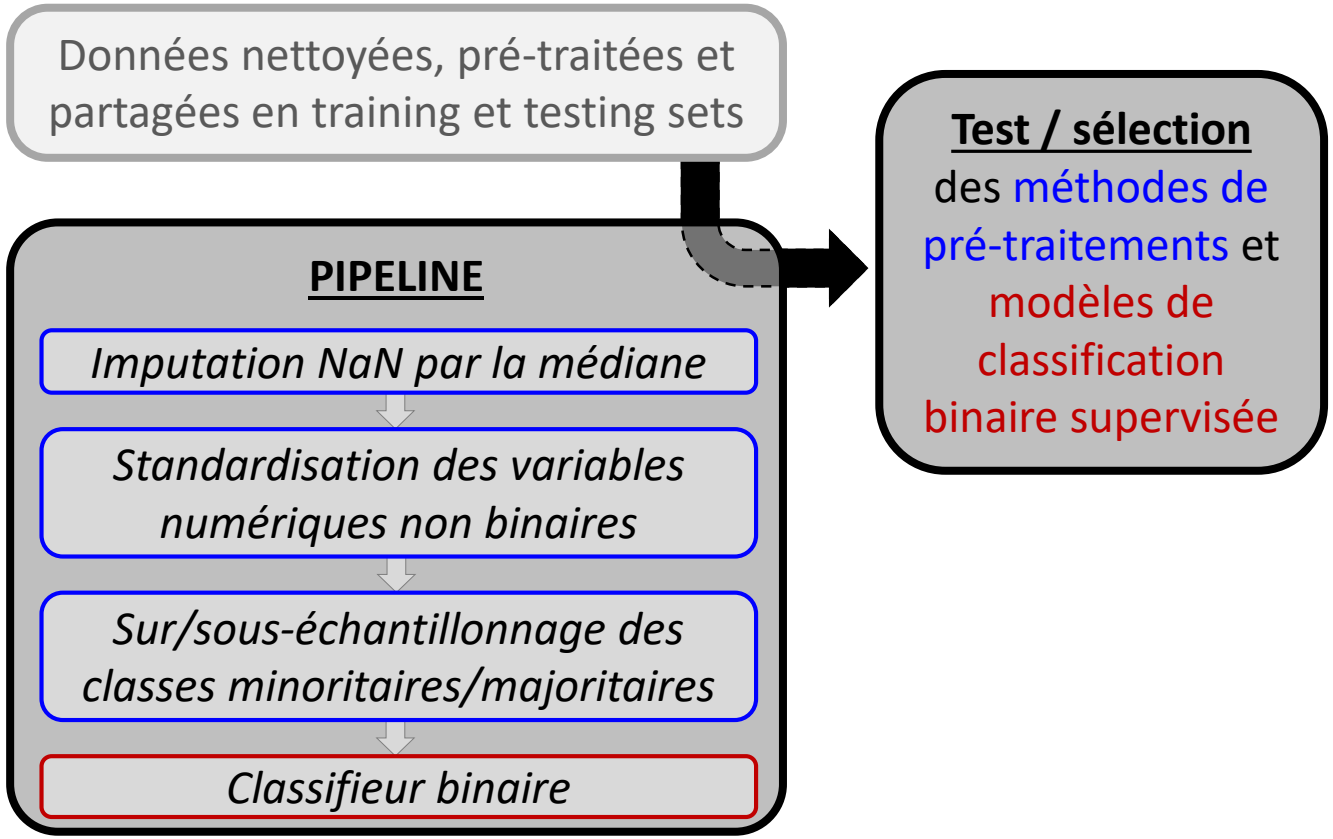
Données nettoyées, pré-traitées et partagées en training et testing sets

PIPELINE

- *Imputation NaN par la médiane*
- *Standardisation des variables numériques non binaires*
- *Sur/sous-échantillonnage des classes minoritaires/majoritaires*
- *Classifieur binaire*

Test / sélection des méthodes de pré-traitements et modèles de classification binaire supervisée

Test et sélection des méthodes de pré-traitements et modèle



Tests effectués sur sous-échantillon stratifié du jeu d'entrainement (30%)

par GridSearchCV

avec métriques ROC-AUC, F1-score et *average precision* (*PR curve*) → performance de prédiction sur la classe minoritaire

Test et sélection des méthodes de pré-traitements et modèle

Données nettoyées, pré-traitées et partagées en training et testing sets

PIPELINE

- *Imputation NaN par la médiane*
- *Standardisation des variables numériques non binaires*
- *Sur/sous-échantillonnage des classes minoritaires/majoritaires*
- *Classifieur binaire*

Test / sélection des méthodes de pré-traitements et modèles de classification binaire supervisée

Méthodes/modèles testés:

- Imputation: ON/OFF
- Méthodes de sur/sous-échantillonnage : SMOTE, RandomUnderSampler
- Modèles : régression logistique (baseline), Random Forest, Gradient Boosting, XGBoost et LightGBM

Tests effectués sur sous-échantillon stratifié du jeu d'entrainement (30%)

par GridSearchCV

avec métriques ROC-AUC, F1-score et *average precision* (*PR curve*) → performance de prédiction sur la classe minoritaire

Test et sélection des méthodes de pré-traitements et modèle

Données nettoyées, pré-traitées et partagées en training et testing sets

PIPELINE

- *Imputation NaN par la médiane*
- *Standardisation des variables numériques non binaires*
- *Sur/sous-échantillonnage des classes minoritaires/majoritaires*
- *Classifieur binaire*

Test / sélection des méthodes de pré-traitements et modèles de classification binaire supervisée

Tests effectués sur sous-échantillon stratifié du jeu d'entrainement (30%)

par GridSearchCV

avec métriques ROC-AUC, F1-score et *average precision* (*PR curve*) → performance de prédiction sur la classe minoritaire

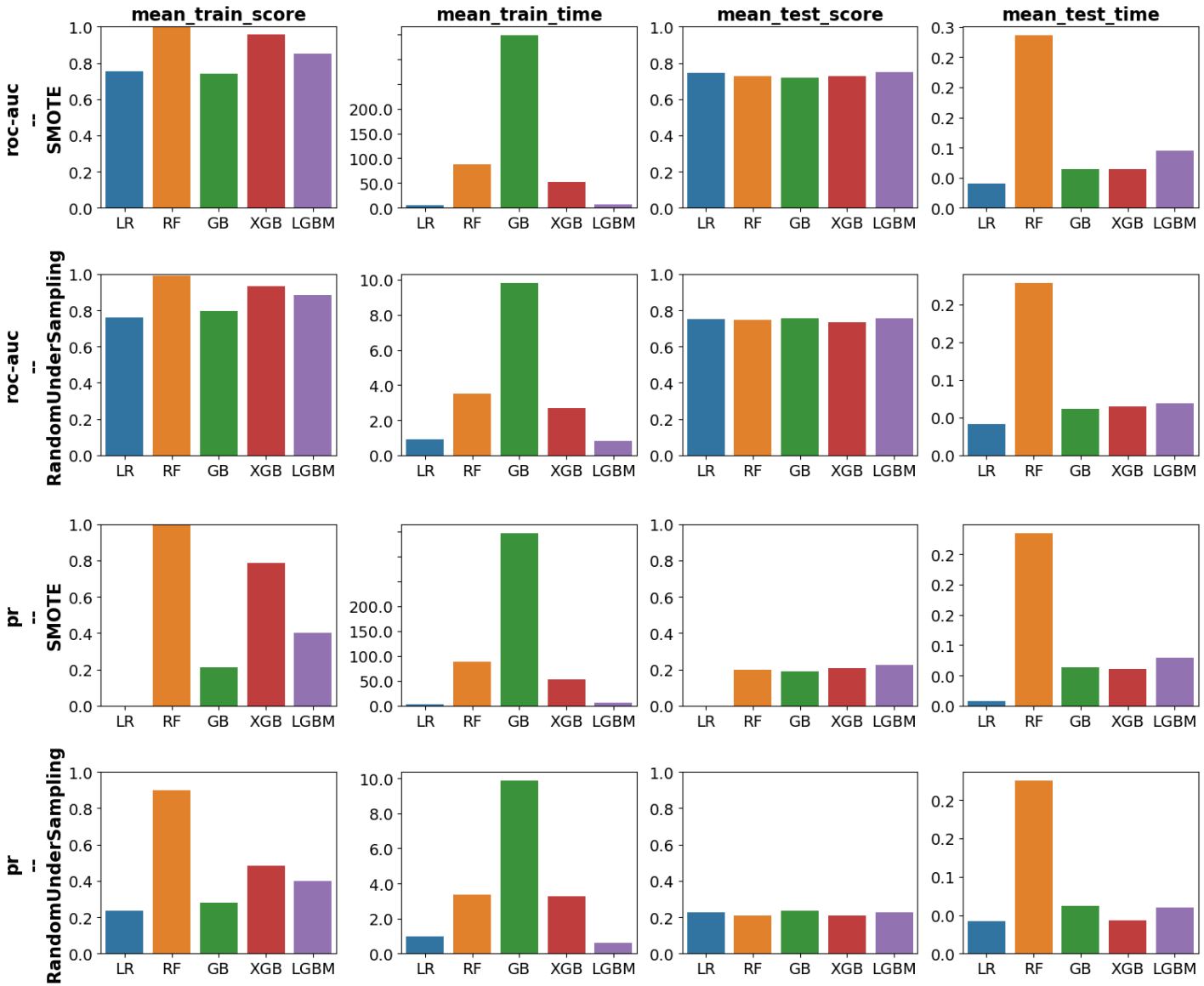
Méthodes/modèles testés:

- Imputation: ON/OFF
- Méthodes de sur/sous-échantillonnage : SMOTE, RandomUnderSampler
- Modèles : régression logistique (baseline), Random Forest, Gradient Boosting, XGBoost et LightGBM

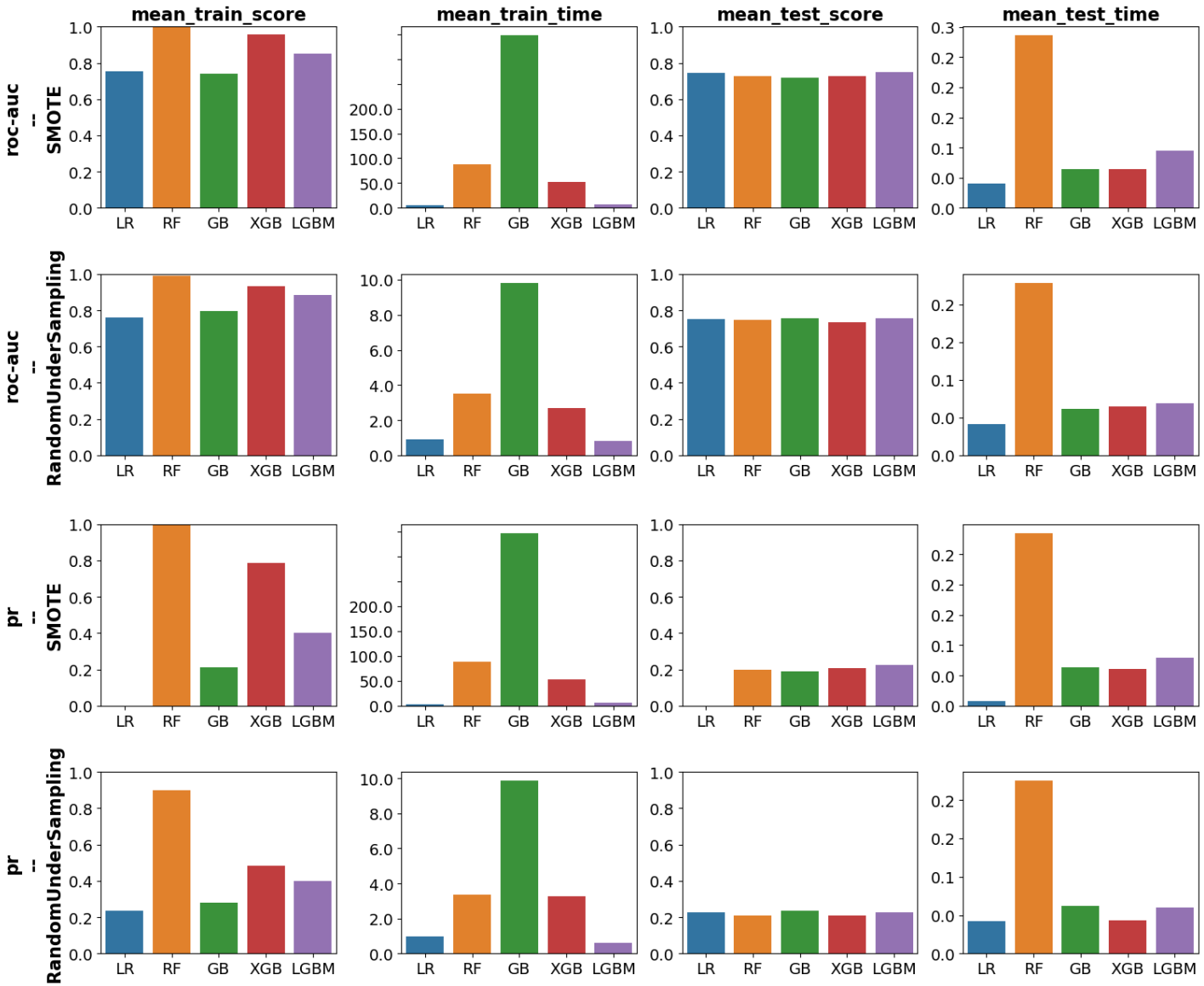
Avec hyperparamètres:

- **valeurs par défaut**
- **SAUF** hyperparamètre du classifieur permettant de prendre en compte le caractère déséquilibré de la variable cible (« class_weight » ou équivalent): ON

Test et sélection des méthodes de pré-traitements et modèle



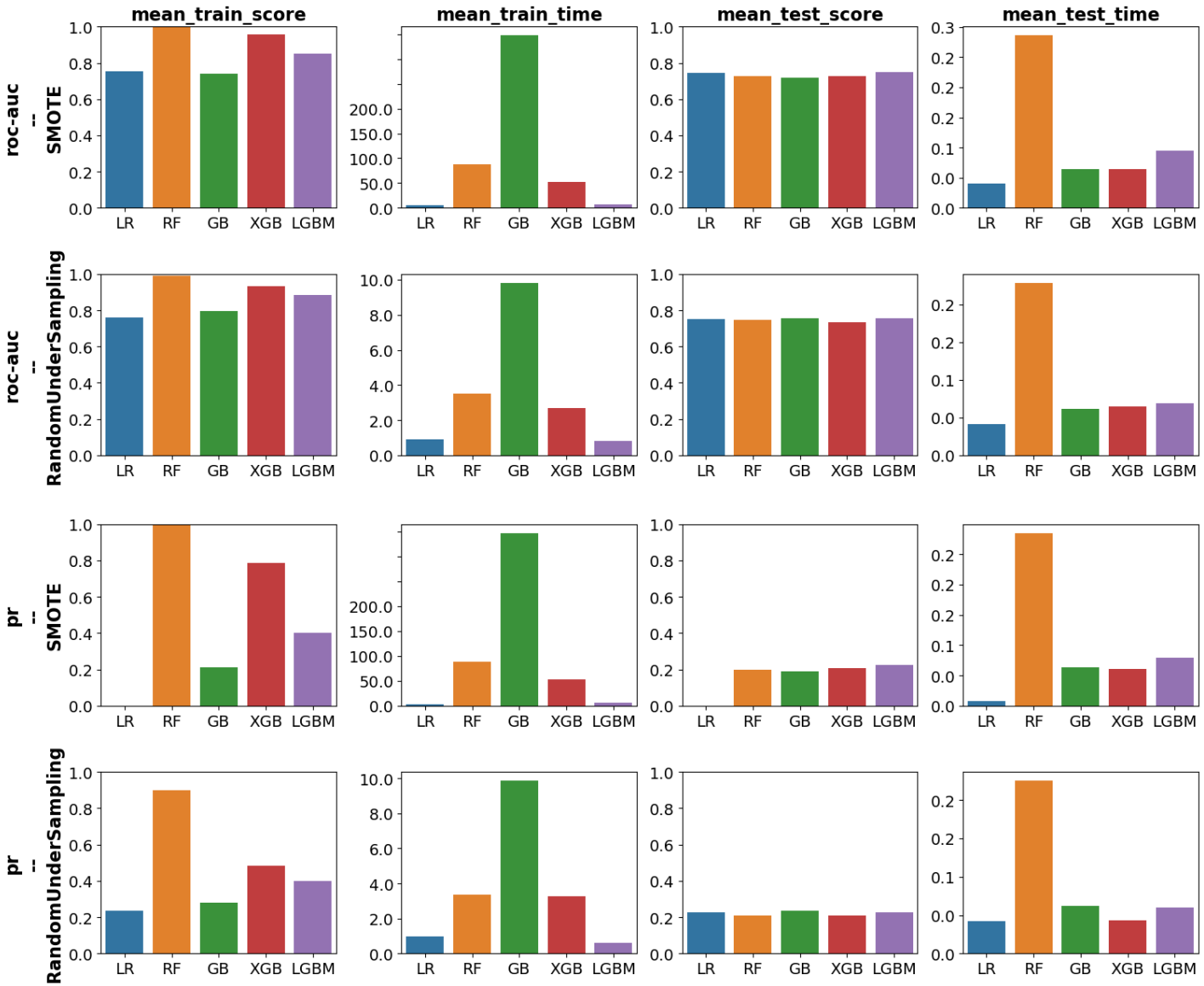
Test et sélection des méthodes de pré-traitements et modèle



Sélection modèle:

- **performances** sur le jeu de validation équivalentes entre les différents modèles… MAIS LightGBM constamment dans TOP2 des modèles les plus performants
- **temps** d'entrainement les plus faibles et temps de prédiction équivalents pour LGBM

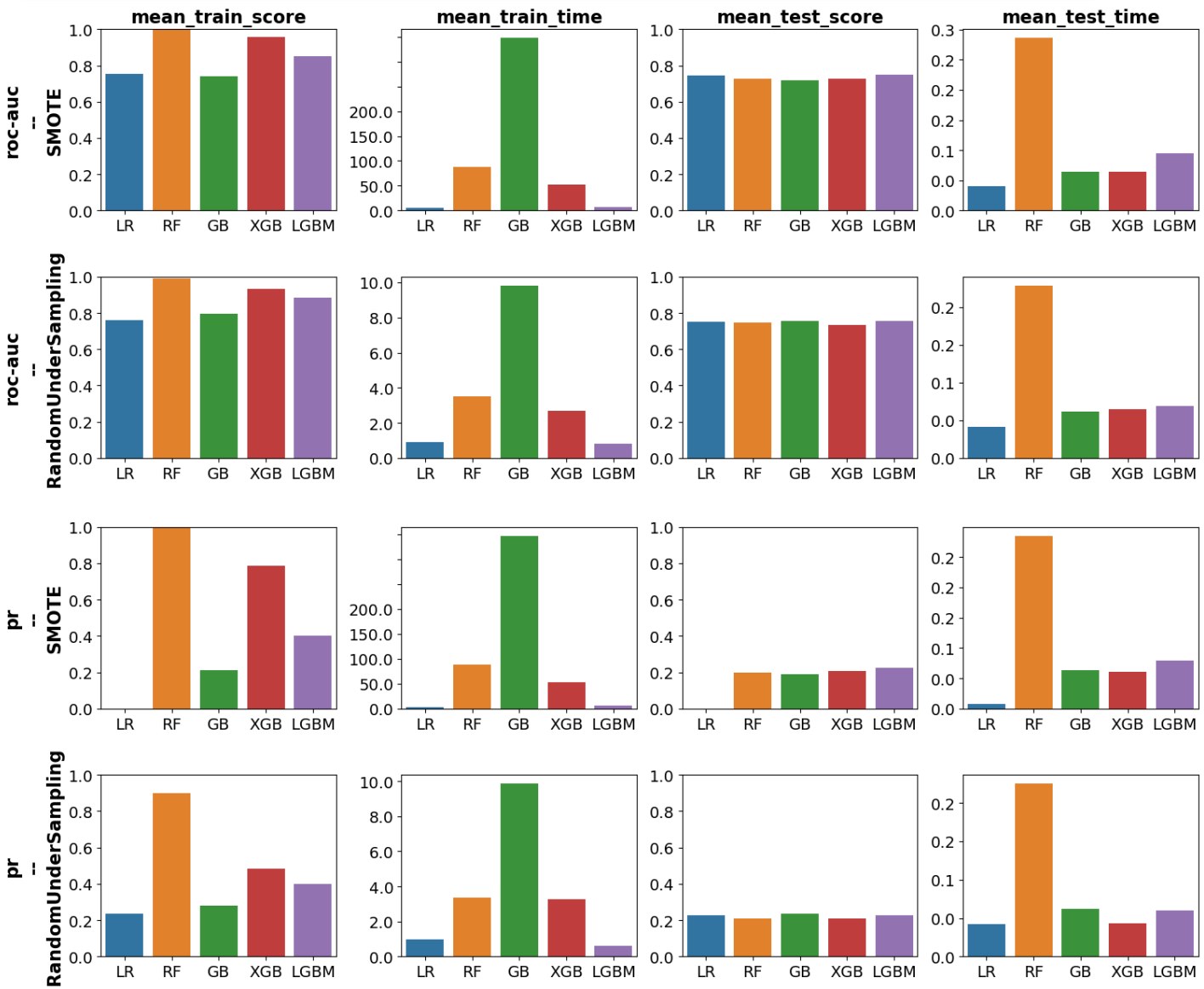
Test et sélection des méthodes de pré-traitements et modèle



Sélection modèle: ➔ **LGBM sélectionné**

- **performances** sur le jeu de validation équivalentes entre les différents modèles… MAIS LightGBM constamment dans TOP2 des modèles les plus performants
- **temps** d'entrainement les plus faibles et temps de prédiction équivalents pour LGBM

Test et sélection des méthodes de pré-traitements et modèle



Sélection modèle: ➔ **LGBM sélectionné**

- **performances** sur le jeu de validation équivalentes entre les différents modèles… MAIS LightGBM constamment dans TOP2 des modèles les plus performants
- **temps** d'entrainement les plus faibles et temps de prédiction équivalents pour LGBM

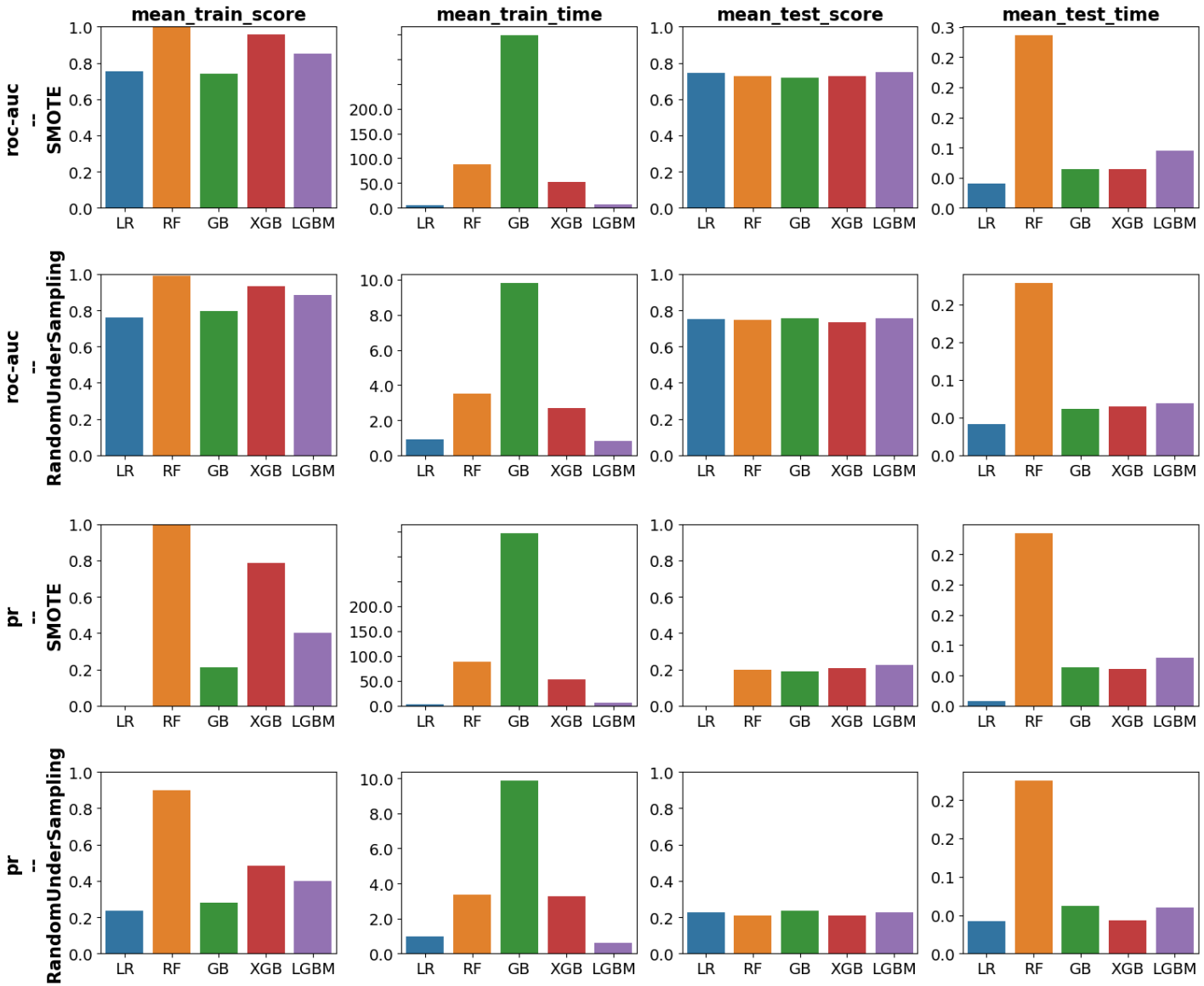
Sélection méthodes de pré-traitement:

à métrique identique, performances constamment meilleures de LGBM lorsque

(1) les données manquantes ne sont pas imputées

(2) les données sont sous-échantillonnées

Test et sélection des méthodes de pré-traitements et modèle



Sélection modèle: ➔ **LGBM sélectionné**

- **performances** sur le jeu de validation équivalentes entre les différents modèles… MAIS LightGBM constamment dans TOP2 des modèles les plus performants
- **temps** d'entrainement les plus faibles et temps de prédiction équivalents pour LGBM

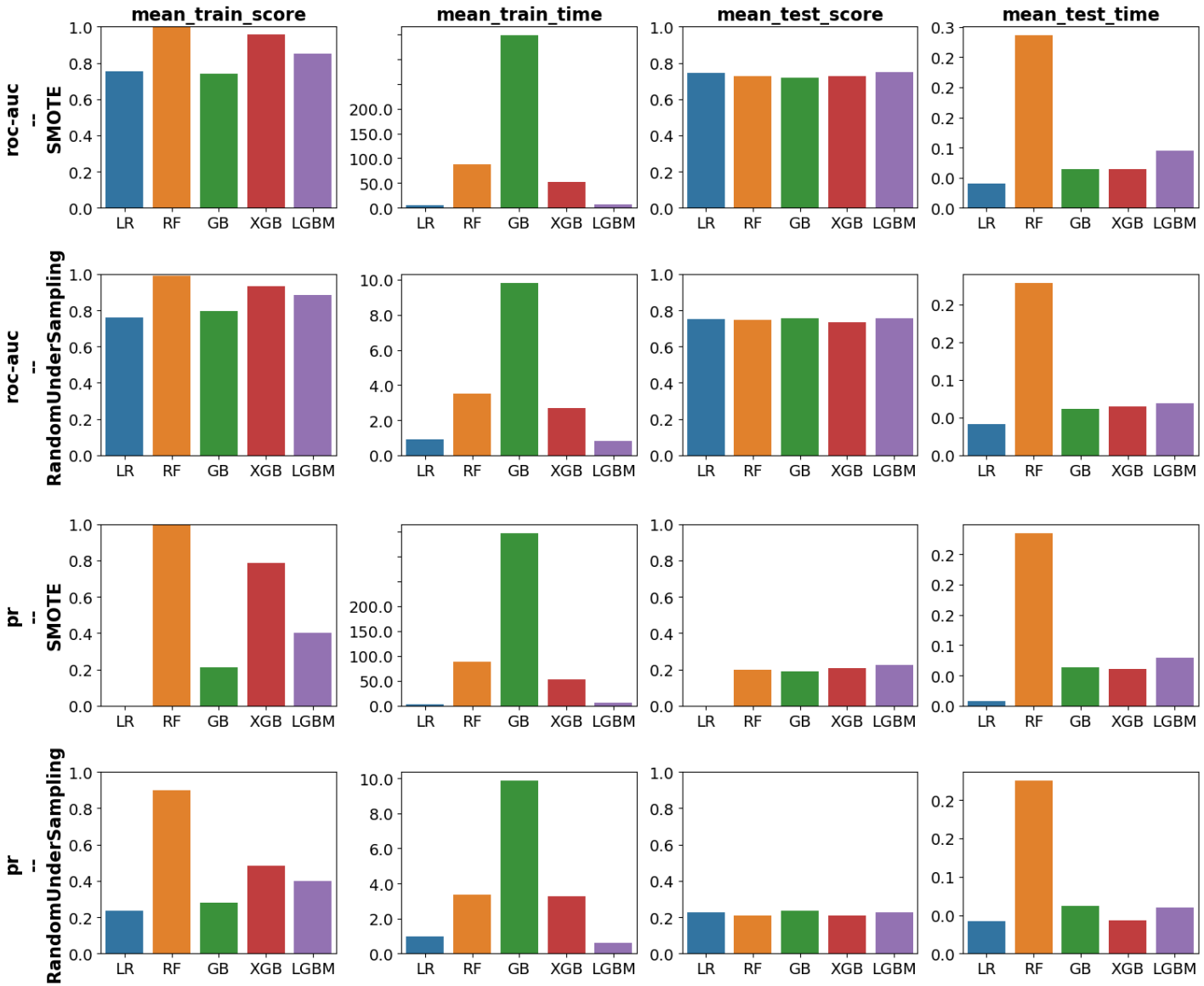
Sélection méthodes de pré-traitement:

à métrique identique, performances constamment meilleures de LGBM lorsque

(1) les données manquantes ne sont pas imputées ➔ **Imputation supprimée**

(2) les données sont sous-échantillonnées

Test et sélection des méthodes de pré-traitements et modèle



Sélection modèle: ➔ **LGBM sélectionné**

- **performances** sur le jeu de validation équivalentes entre les différents modèles… MAIS LightGBM constamment dans TOP2 des modèles les plus performants
- **temps** d'entrainement les plus faibles et temps de prédiction équivalents pour LGBM

Sélection méthodes de pré-traitement:

à métrique identique, performances constamment meilleures de LGBM lorsque

(1) les données manquantes ne sont pas imputées ➔ **Imputation supprimée**

(2) les données sont sous-échantillonnées ➔ **RandomUnderSampler() sélectionné**

Optimisation des hyperparamètres

Données nettoyées, pré-traitées et partagées en training et testing sets

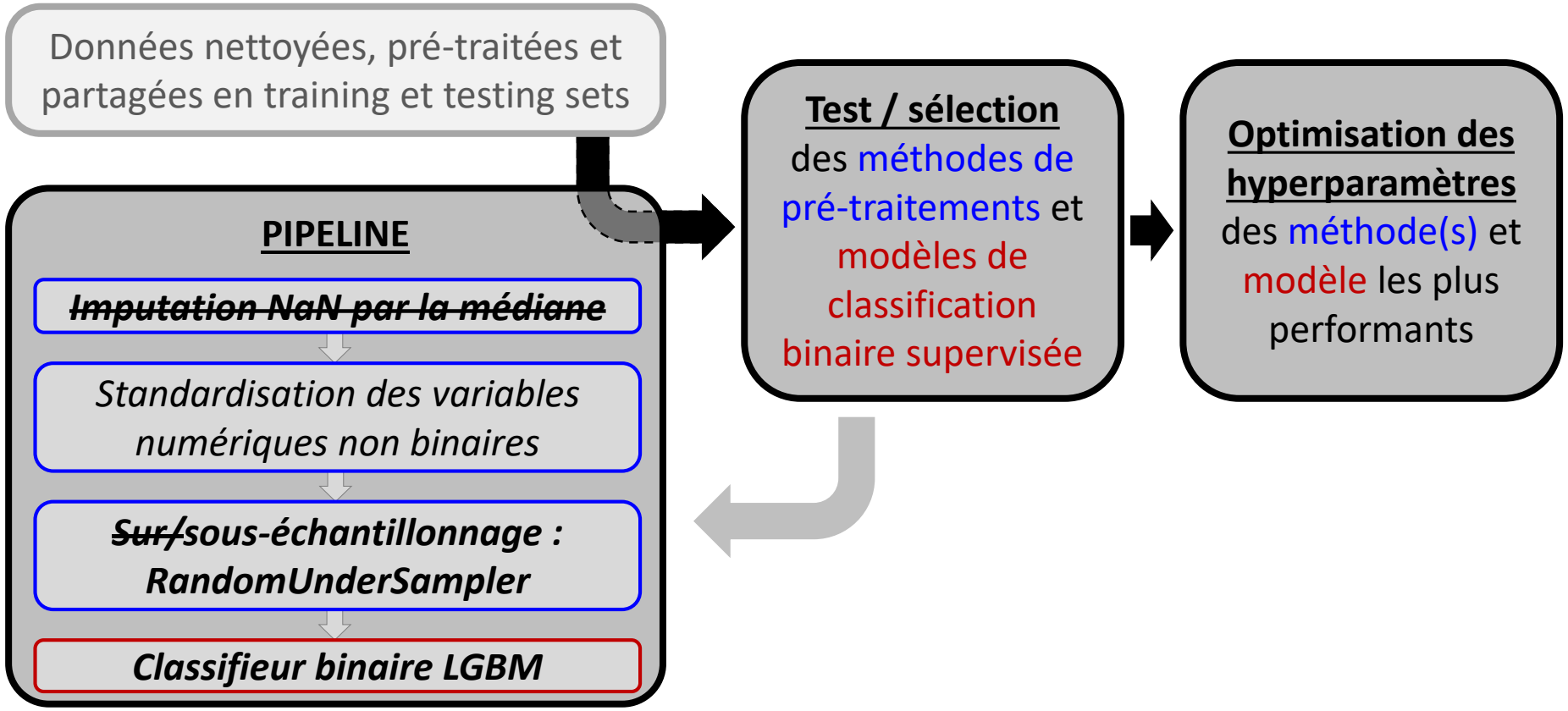
PIPELINE

- *Imputation NaN par la médiane*
- *Standardisation des variables numériques non binaires*
- *Sur/sous-échantillonnage des classes minoritaires/majoritaires*
- *Classifieur binaire*

Test / sélection des méthodes de pré-traitements et modèles de classification binaire supervisée

Optimisation des hyperparamètres des méthode(s) et modèle les plus performants

Optimisation des hyperparamètres



Optimisation des hyperparamètres

Données nettoyées, pré-traitées et partagées en training et testing sets

PIPELINE

- ~~*Imputation NaN par la médiane*~~
- *Standardisation des variables numériques non binaires*
- ~~*Sur-*~~*sous-échantillonnage : RandomUnderSampler*
- *Classifieur binaire LGBM*

Test / sélection des méthodes de pré-traitements et modèles de classification binaire supervisée

Optimisation des hyperparamètres des méthode(s) et modèle les plus performants

	Hyperparamètre	Gamme testée
RUS	sampling_strategy	0.1 à 0.9
LightGBM	n_estimators	100 à 1000
	learning_rate	0.01 à 0.3
	max_depth	3 à 12
	num_leaves	20 à 1000
	colsample_bytree	0.8 à 1
	subsample	0.8 à 1
	reg_alpha	0.001 à 10
	reg_lambda	0.001 à 10

Effectuée sur totalité du jeu d'entrainement

par OptunaSearchCV (approche bayésienne)

avec métrique *average precision* (*PR curve*) → performance de prédiction sur la classe minoritaire SEULEMENT

Optimisation des hyperparamètres

Données nettoyées, pré-traitées et partagées en training et testing sets

PIPELINE

- ~~*Imputation NaN par la médiane*~~
- *Standardisation des variables numériques non binaires*
- ~~*Sur-*~~*sous-échantillonnage : RandomUnderSampler*
- *Classifieur binaire LGBM*

Test / sélection des méthodes de pré-traitements et modèles de classification binaire supervisée

Optimisation des hyperparamètres des méthode(s) et modèle les plus performants

	Hyperparamètre	Gamme testée	Valeur optimale
RUS	sampling_strategy	0.1 à 0.9	0.10327
LightGBM	n_estimators	100 à 1000	641
	learning_rate	0.01 à 0.3	0.07070
	max_depth	3 à 12	3
	num_leaves	20 à 1000	811
	colsample_bytree	0.8 à 1	0.96894
	subsample	0.8 à 1	0.98835
	reg_alpha	0.001 à 10	0.00415
	reg_lambda	0.001 à 10	0.04095

Effectuée sur totalité du jeu d'entrainement

par OptunaSearchCV (approche bayésienne)

avec métrique *average precision* (*PR curve*) → performance de prédiction sur la classe minoritaire SEULEMENT

Optimisation des hyperparamètres

Données nettoyées, pré-traitées et partagées en training et testing sets

PIPELINE

- ~~*Imputation NaN par la médiane*~~
- *Standardisation des variables numériques non binaires*
- ~~*Sur-*~~*sous-échantillonnage : RandomUnderSampler*
- *Classifieur binaire LGBM*

Test / sélection des méthodes de pré-traitements et modèles de classification binaire supervisée

Optimisation des hyperparamètres des méthode(s) et modèle les plus performants

	Hyperparamètre	Gamme testée	Valeur optimale
RUS	sampling_strategy	0.1 à 0.9	0.10327
LightGBM	n_estimators	100 à 1000	641
	learning_rate	0.01 à 0.3	0.07070
	max_depth	3 à 12	3
	num_leaves	20 à 1000	811
	colsample_bytree	0.8 à 1	0.96894
	subsample	0.8 à 1	0.98835
	reg_alpha	0.001 à 10	0.00415
	reg_lambda	0.001 à 10	0.04095

Effectuée sur totalité du jeu d'entrainement

par OptunaSearchCV (approche bayésienne)

avec métrique *average precision* (*PR curve*) → performance de prédiction sur la classe minoritaire SEULEMENT

↪ **0,23 → 0,26**

Optimisation des hyperparamètres

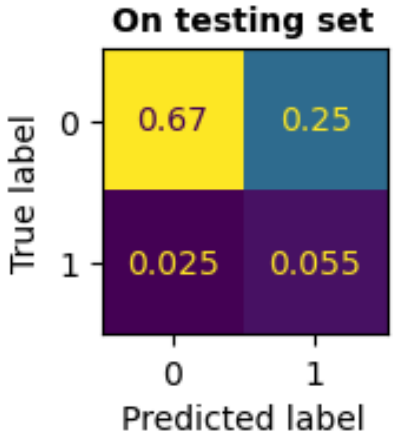
Données nettoyées, pré-traitées et partagées en training et testing sets

PIPELINE

- ~~*Imputation NaN par la médiane*~~
- *Standardisation des variables numériques non binaires*
- ~~*Sur-*~~*sous-échantillonnage : RandomUnderSampler*
- *Classifieur binaire LGBM*

Test / sélection des méthodes de pré-traitements et modèles de classification binaire supervisée

Optimisation des hyperparamètres des méthode(s) et modèle les plus performants



Effectuée sur totalité du jeu d'entrainement

par OptunaSearchCV (approche bayésienne)

avec métrique *average precision* (*PR curve*) → performance de prédiction sur la classe minoritaire SEULEMENT

↪ **0,23 → 0,26**

Optimisation des hyperparamètres

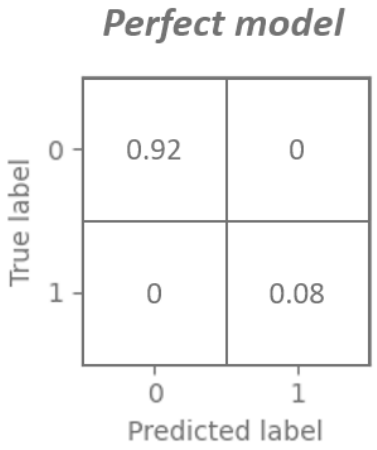
Données nettoyées, pré-traitées et partagées en training et testing sets

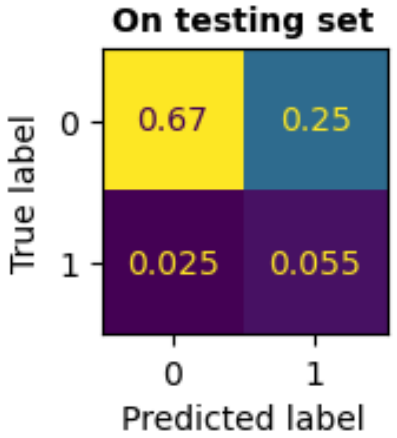
PIPELINE

- ~~*Imputation NaN par la médiane*~~
- *Standardisation des variables numériques non binaires*
- ~~*Sur-*~~*sous-échantillonnage : RandomUnderSampler*
- *Classifieur binaire LGBM*

Test / sélection des méthodes de pré-traitements et modèles de classification binaire supervisée

Optimisation des hyperparamètres des méthode(s) et modèle les plus performants





Effectuée sur totalité du jeu d'entrainement

par OptunaSearchCV (approche bayésienne)

avec métrique *average precision* (*PR curve*) → performance de prédiction sur la classe minoritaire SEULEMENT

↪ **0,23 → 0,26**

Optimisation du seuil

Données nettoyées, pré-traitées et partagées en training et testing sets

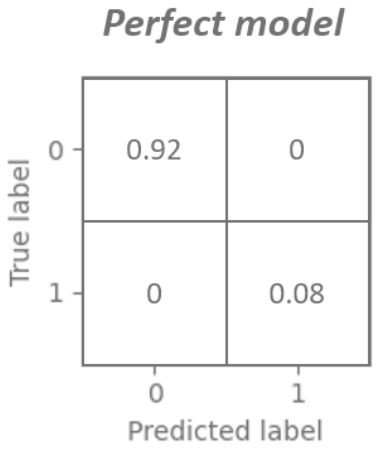
PIPELINE

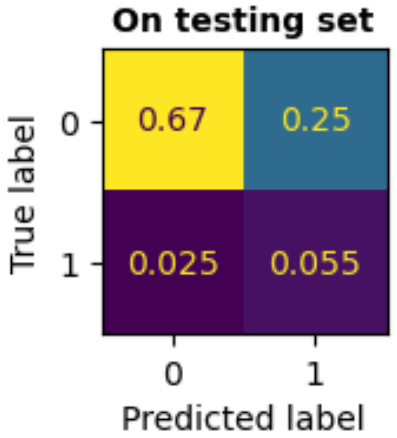
- ~~*Imputation NaN par la médiane*~~
- *Standardisation des variables numériques non binaires*
- ~~*Sur-*~~*sous-échantillonnage : RandomUnderSampler*
- *Classifieur binaire LGBM*

Test / sélection des méthodes de pré-traitements et modèles de classification binaire supervisée

Optimisation des hyperparamètres des méthode(s) et modèle les plus performants

Optimisation du seuil de probabilité conditionnant l'attribution des classes





Optimisation du seuil

Données nettoyées, pré-traitées et partagées en training et testing sets

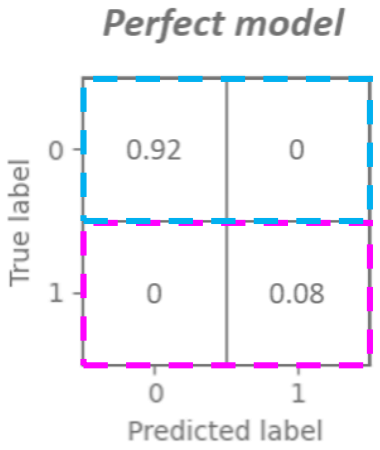
PIPELINE

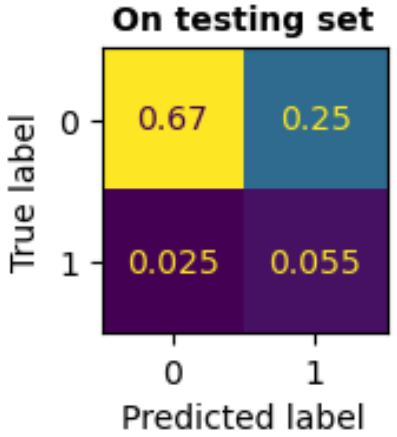
- ~~*Imputation NaN par la médiane*~~
- *Standardisation des variables numériques non binaires*
- ~~*Sur-*~~*sous-échantillonnage : RandomUnderSampler*
- *Classifieur binaire LGBM*

Test / sélection des méthodes de pré-traitements et modèles de classification binaire supervisée

Optimisation des hyperparamètres des méthode(s) et modèle les plus performants

Optimisation du seuil de probabilité conditionnant l'attribution des classes





modèle plus performant dans l'identification des vrais mauvais payeurs (au détriment de celle des bons payeurs)

Optimisation du seuil

Données nettoyées, pré-traitées et partagées en training et testing sets

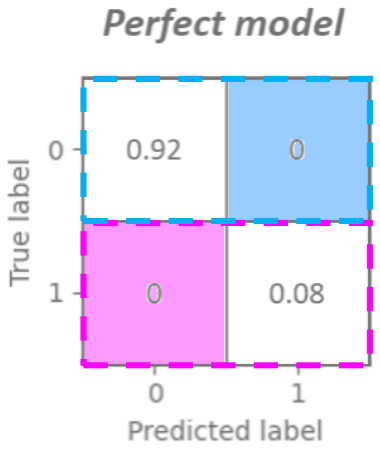
PIPELINE

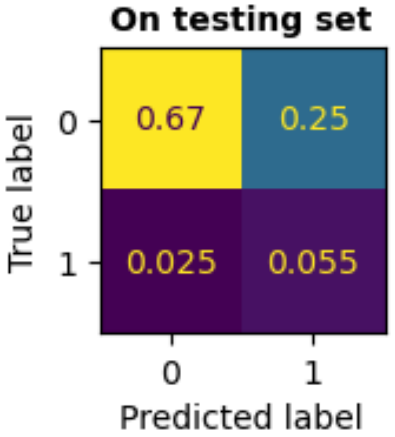
- ~~*Imputation NaN par la médiane*~~
- *Standardisation des variables numériques non binaires*
- ~~*Sur-*~~*sous-échantillonnage : RandomUnderSampler*
- *Classifieur binaire LGBM*

Test / sélection des méthodes de pré-traitements et modèles de classification binaire supervisée

Optimisation des hyperparamètres des méthode(s) et modèle les plus performants

Optimisation du seuil de probabilité conditionnant l'attribution des classes





modèle plus performant dans l'identification des vrais mauvais payeurs (au détriment de celle des bons payeurs)

=

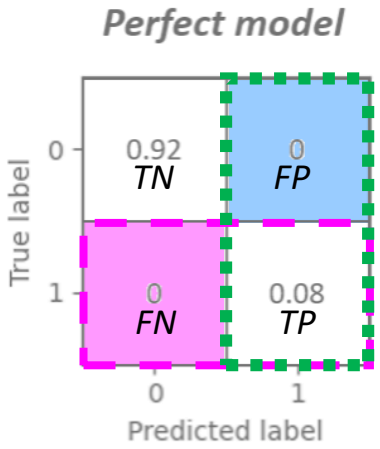
refuse relativement plus de crédits à tort qu'il n'en accorde à tort

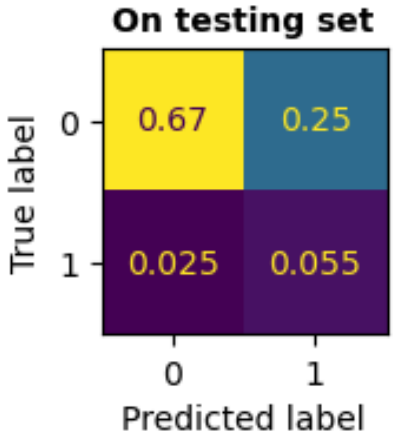
Optimisation du seuil

$$F_\beta = (1+\beta^2) \cdot \frac{\frac{TP}{TP+FP} \cdot \frac{TP}{TP+FN}}{\left(\beta^2 \cdot \frac{TP}{TP+FP}\right) + \frac{TP}{TP+FN}}$$

Optimisation des hyperparamètres des méthode(s) et modèle les plus performants → **Optimisation du seuil de probabilité** conditionnant l'attribution des classes

Taux de vrais positifs (rappel) β fois plus important que proportion de prédictions positives correctes (précision)



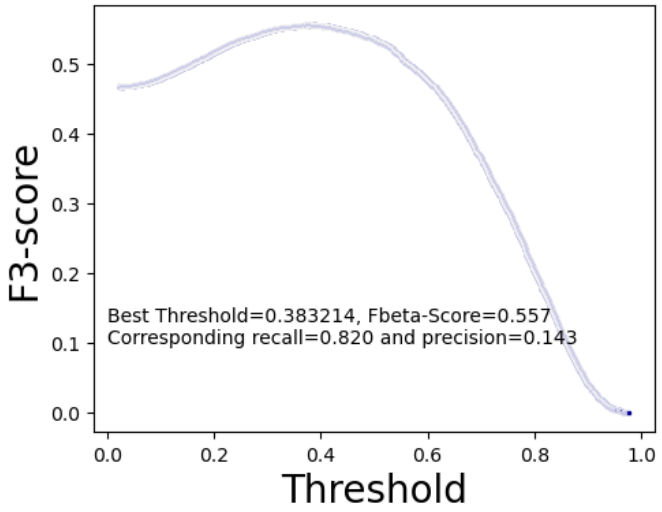


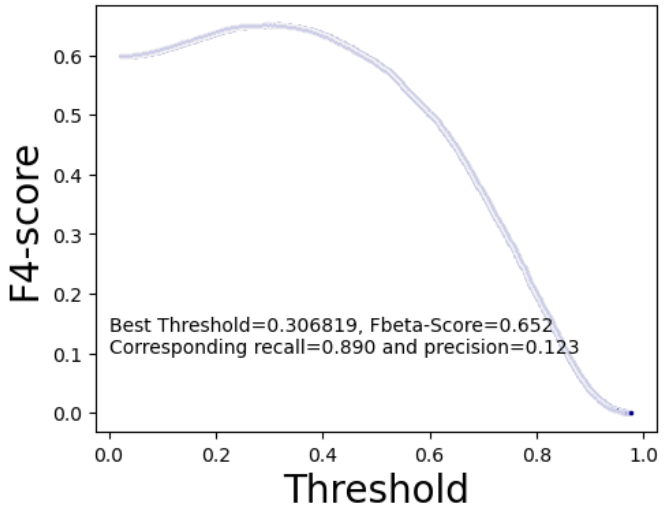
modèle plus performant dans l'identification des vrais mauvais payeurs (au détriment de celle des bons payeurs)

=

refuse relativement plus de crédits à tort qu'il n'en accorde à tort

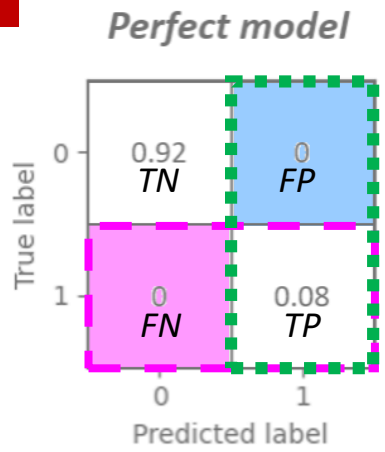
Optimisation du seuil





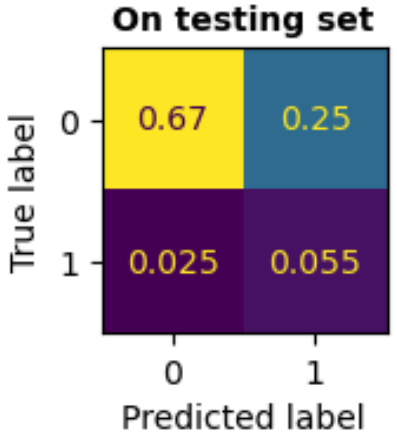
Plusieurs valeurs de β testées
➔ compromis: seuil = 0,35

Taux de vrais positifs (rappel) β fois plus important que proportion de prédictions positives correctes (précision)



<u>Optimisation des hyperparamètres</u> des méthode(s) et modèle les plus performants

<u>Optimisation du seuil de probabilité</u> conditionnant l'attribution des classes



modèle plus performant dans l'identification des vrais mauvais payeurs (au détriment de celle des bons payeurs)

=

refuse relativement plus de crédits à tort qu'il n'en accorde à tort

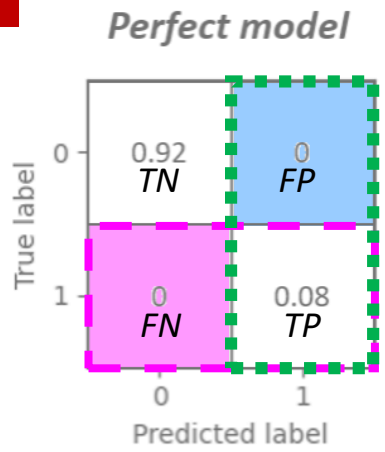
Optimisation du seuil

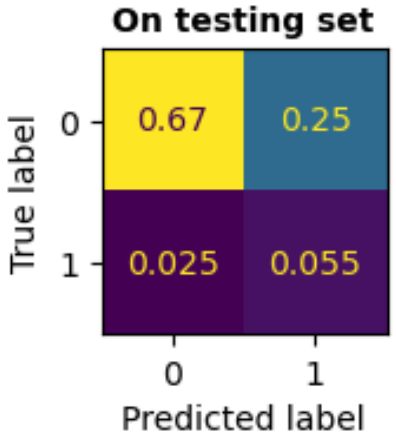
Optimisation des hyperparamètres des méthode(s) et modèle les plus performants

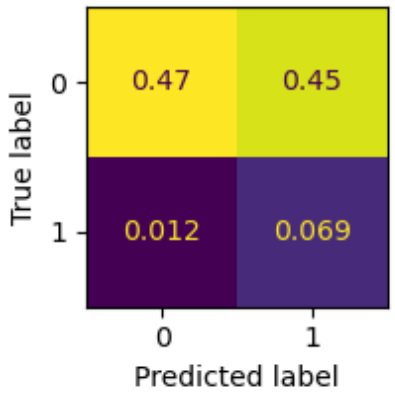
Optimisation du seuil de probabilité conditionnant l'attribution des classes

Plusieurs valeurs de β testées
➔ compromis: seuil = 0,35

➔ Seuls 15% (contre 31% avant) des vrais mauvais payeurs sont mal identifiés… mais en contrepartie 87% (82% avant) des prédictions de mauvais payeurs sont en réalité de bons payeurs.







Interprétation

Données nettoyées, pré-traitées et partagées en training et testing sets

PIPELINE

- ~~*Imputation NaN par la médiane*~~
- *Standardisation des variables numériques non binaires*
- ~~*Sur-*~~*sous-échantillonnage : RandomUnderSampler*
- *Classifieur binaire LGBM*

Test / sélection des méthodes de pré-traitements et modèles de classification binaire supervisée

Optimisation des hyperparamètres des méthode(s) et modèle les plus performants

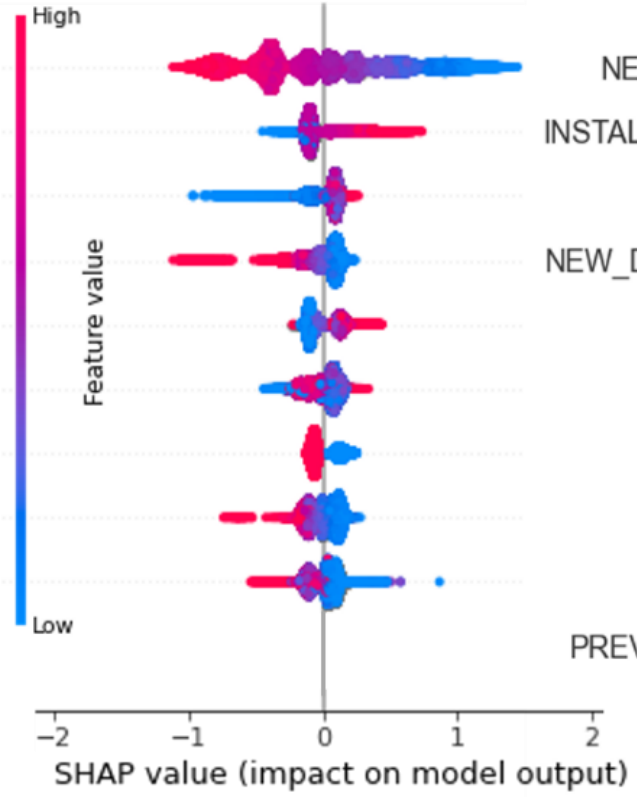
Optimisation du seuil de probabilité conditionnant l'attribution des classes

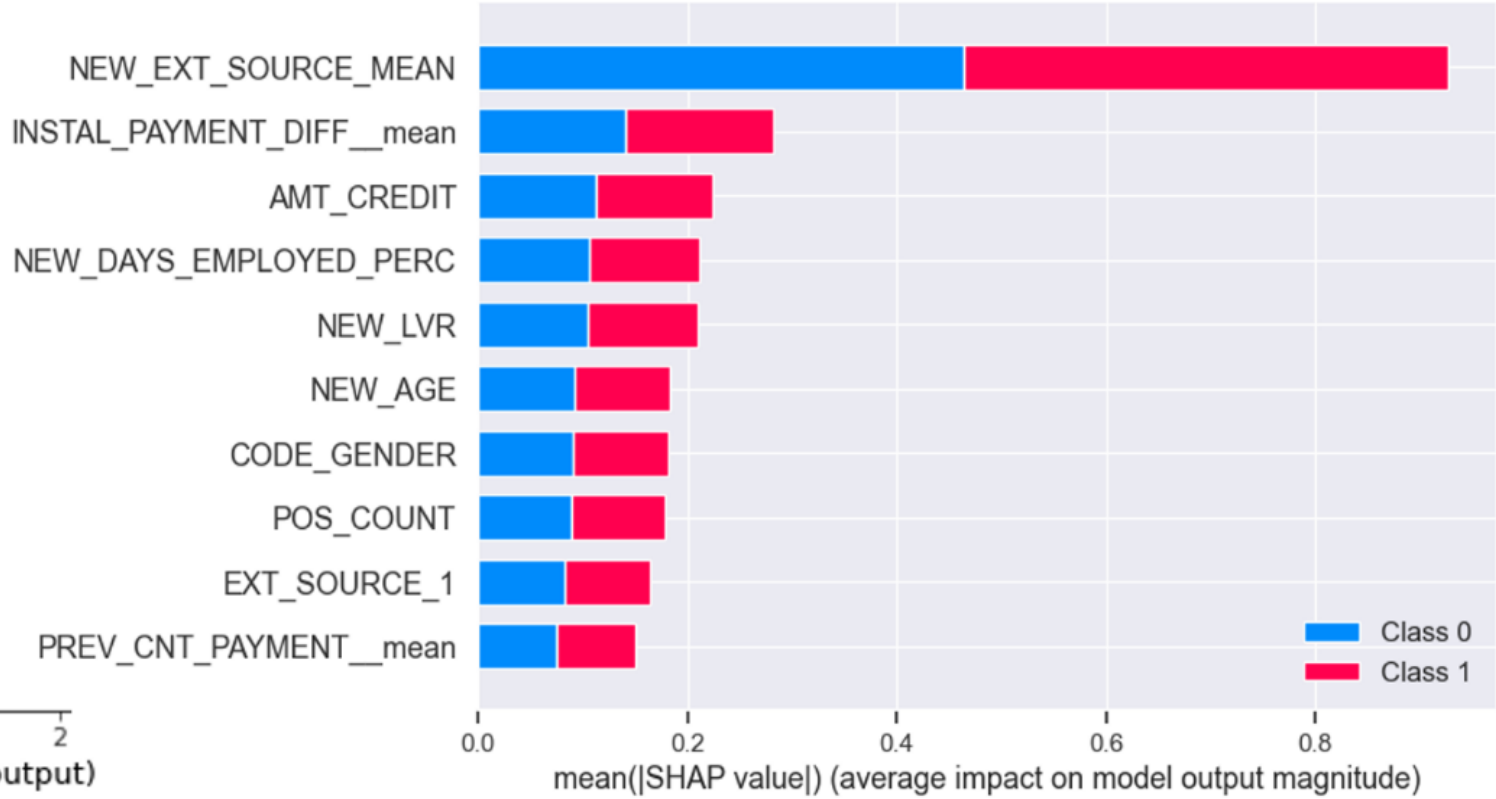
Interprétation globale et locale des prédictions du modèle

SHAP

Interprétation globale

SHAP





VARIABLES CONTRIBUANT LE PLUS AUX PREDICTIONS DU MODELE:

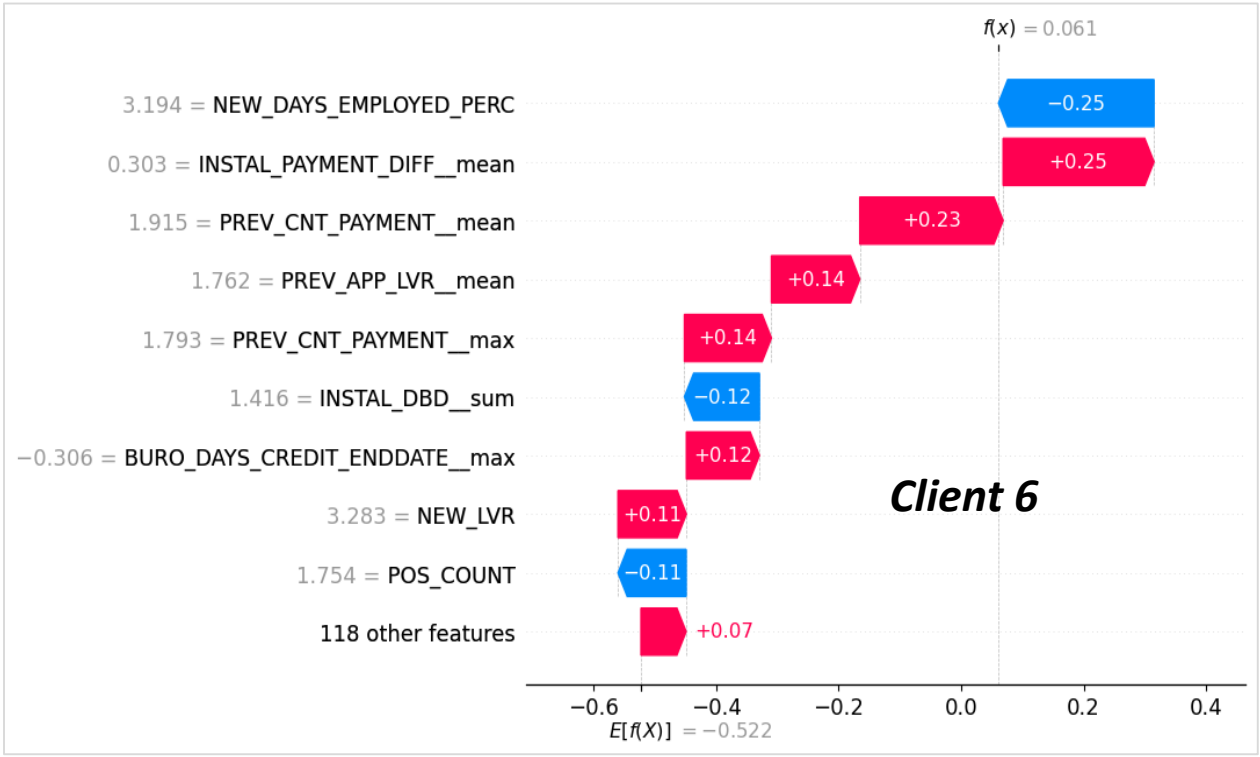
1. *moyenne des scores précédemment attribués aux clients par d'autres banques (NEW_EXT_SOURCE_MEAN)*
2. *différence moyenne entre montant dû pour chaque mensualité et montant correspondant réellement remboursé sur les précédents crédits des clients ('INSTAL_PAYMENT_DIFF__mean')*
3. *montant du crédit actuellement demandé ('AMT_CREDIT'),*
4. *durée d'emploi relative des clients (par rapport à leur âge – 'NEW_DAYS_EMPLOYED_PERC'),*
5. *ratio entre montant du crédit demandé et montant du bien à acquérir avec ('NEW_LVR' ; LVR pour Loan to Value Ratio),*
6. *âge des clients ('NEW_AGE'),*
7. *genre des clients ('CODE_GENDER'), …*

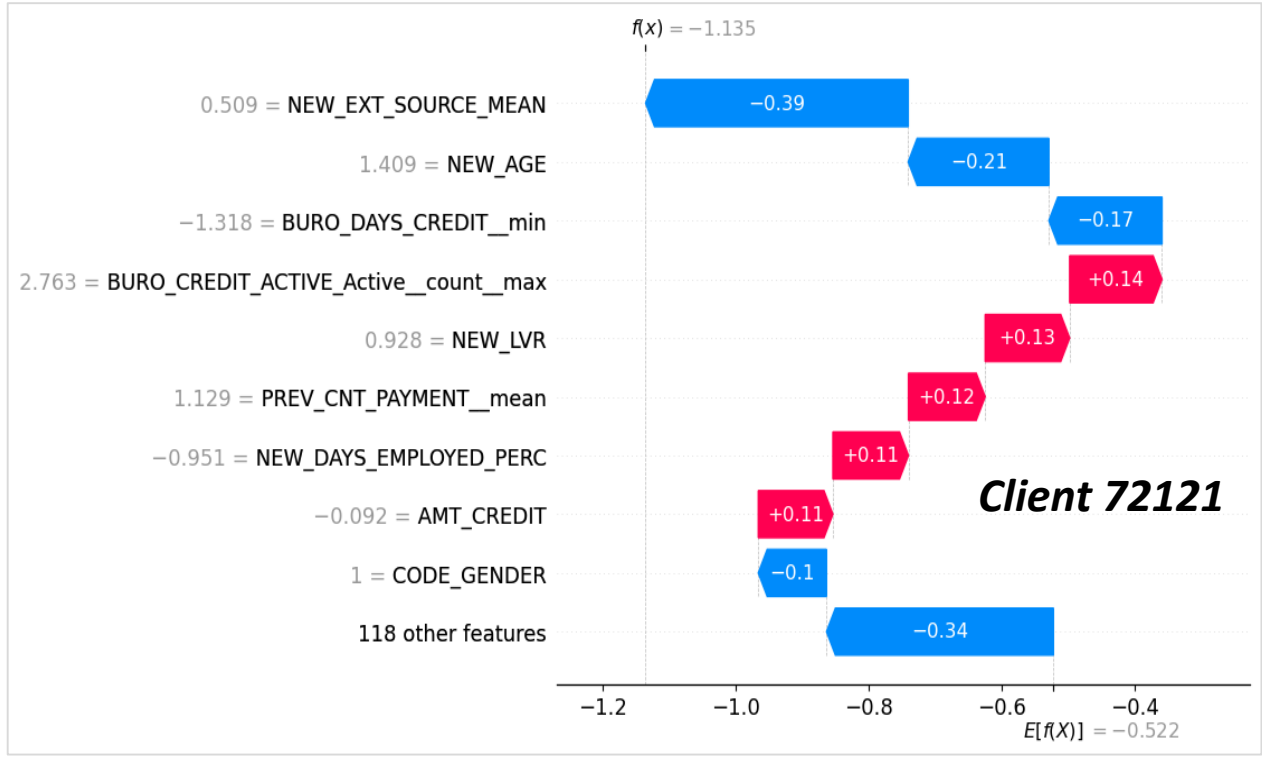
+ **contributions pour la plupart globalement linéaires** (inversement ou non), **mais exceptions où relation non linéaire** (e.g. âge des clients)

Interprétation locale

SHAP

Ces variables qui contribuent le plus au niveau global apparaissent **souvent, mais pas constamment, ni forcément dans le même ordre ou avec la même magnitude d'impact**, parmi les variables influençant le plus le score retourné par le modèle pour chaque individu





Interprétation

Données nettoyées, pré-traitées et partagées en training et testing sets

PIPELINE

- ~~*Imputation NaN par la médiane*~~
- *Standardisation des variables numériques non binaires*
- ~~*Sur-*~~*sous-échantillonnage : RandomUnderSampler*
- *Classifieur binaire LGBM*

Test / sélection des méthodes de pré-traitements et modèles de classification binaire supervisée

Optimisation des hyperparamètres des méthode(s) et modèle les plus performants

Optimisation du seuil de probabilité conditionnant l'attribution des classes

Interprétation globale et locale des prédictions du modèle

SHAP

Déploiement - API

Données nettoyées, pré-traitées et partagées en training et testing sets

PIPELINE

- *~~Imputation NaN par la médiane~~*
- *Standardisation des variables numériques non binaires*
- *~~Sur-~~sous-échantillonnage : RandomUnderSampler*
- *Classifieur binaire LGBM*

Test / sélection des méthodes de pré-traitements et modèles de classification binaire supervisée

Optimisation des hyperparamètres des méthode(s) et modèle les plus performants

Optimisation du seuil de probabilité conditionnant l'attribution des classes

Interprétation globale et locale des prédictions du modèle

Déploiement du modèle sous forme d'API

FastAPI

heroku

Déploiement - API

Données nettoyées, pré-traitées et partagées en training et testing sets

PIPELINE optimisé

Standardisation var. non binaires

RandomUnderSampler

Classifieur binaire LGBM

Test / sélection des méthodes de pré-traitements et modèles de classification binaire supervisée

Optimisation des hyperparamètres des méthode(s) et modèle les plus performants

Optimisation du seuil de probabilité conditionnant l'attribution des classes

Déploiement du modèle sous forme d'API

FastAPI

heroku

Déploiement - API

Données nettoyées, pré-traitées et partagées ~~en training et~~ **testing set**s

PIPELINE optimisé

Standardisation var. non binaires

RandomUnderSampler

Classifieur binaire LGBM

Déploiement du modèle sous forme d'API

FastAPI

heroku

Déploiement - API

Données nettoyées, pré-traitées et partagées ~~en training et~~ **testing set**s

PIPELINE optimisé

Standardisation var. non binaires

RandomUnderSampler

Classifieur binaire LGBM

Jeu de test

Meilleur modèle optimisé

GitHub

Déploiement du modèle sous forme d'API

FastAPI

heroku

Déploiement - API

Données nettoyées, pré-traitées et partagées ~~en training et~~ **testing set**s

PIPELINE optimisé

- *Standardisation var. non binaires*
- *RandomUnderSampler*
- *Classifieur binaire LGBM*

- Jeu de test
- Code API — FastAPI
- Meilleur modèle optimisé

GitHub

Déploiement du modèle sous forme d'API

FastAPI heroku

Déploiement - API

Données nettoyées, pré-traitées et partagées ~~en training et~~ **testing sets**

PIPELINE optimisé

- *Standardisation var. non binaires*
- *RandomUnderSampler*
- *Classifieur binaire LGBM*

- Jeu de test
- Code API — FastAPI
- Meilleur modèle optimisé

heroku

GitHub

Déploiement du modèle sous forme d'API

FastAPI heroku

Déploiement - API

Données nettoyées, pré-traitées et partagées ~~en training et~~ **testing sets**

PIPELINE optimisé

- *Standardisation var. non binaires*
- *RandomUnderSampler*
- *Classifieur binaire LGBM*

- Jeu de test
- Code API — FastAPI
- Meilleur modèle optimisé

heroku

GitHub

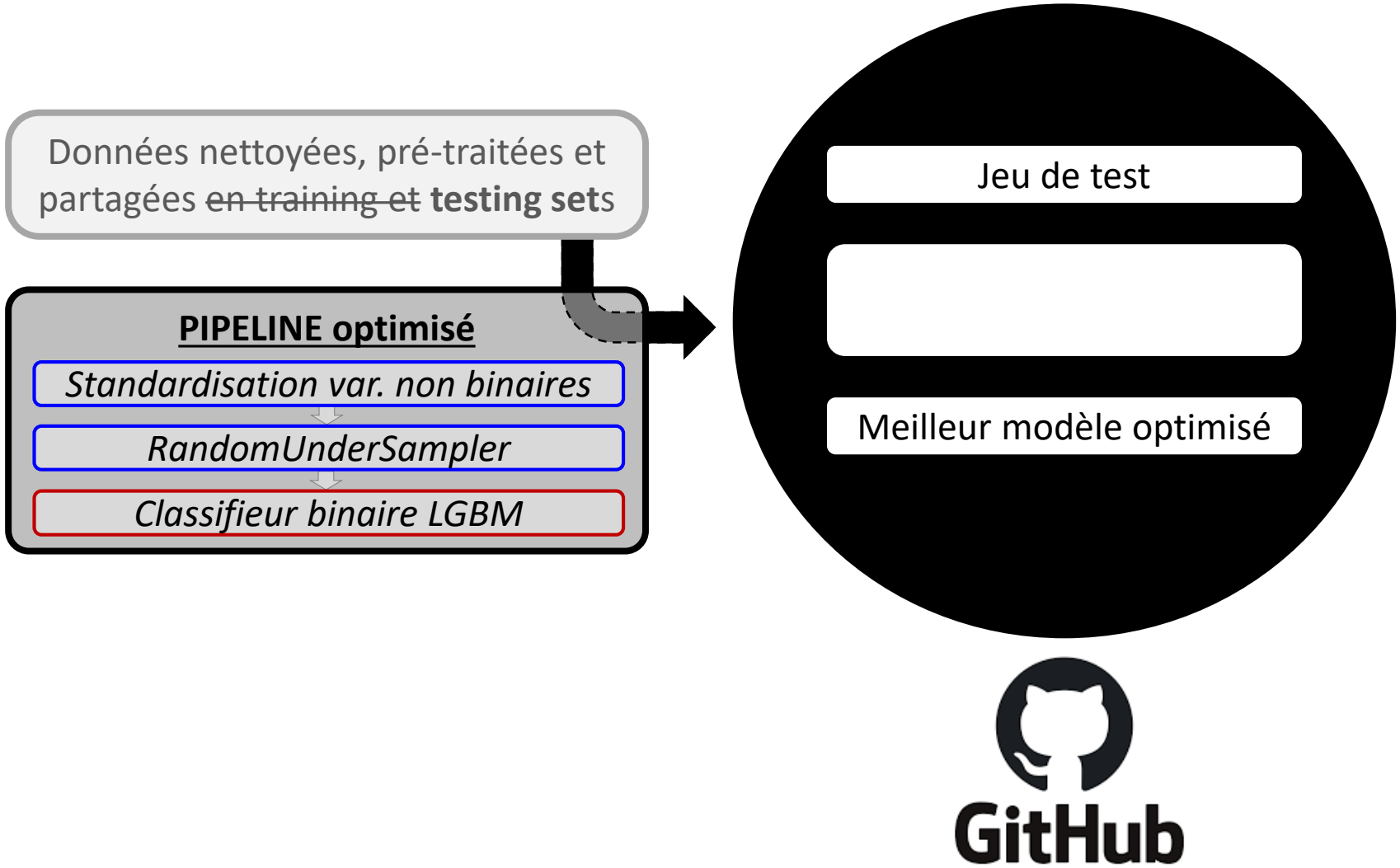
```
import requests
import json
url = 'https://mw-loan-pred.herokuapp.com/prediction'
myobj = {'IDclient': 103676}
x = requests.post(url, data = json.dumps(myobj))
print(x.text)
```

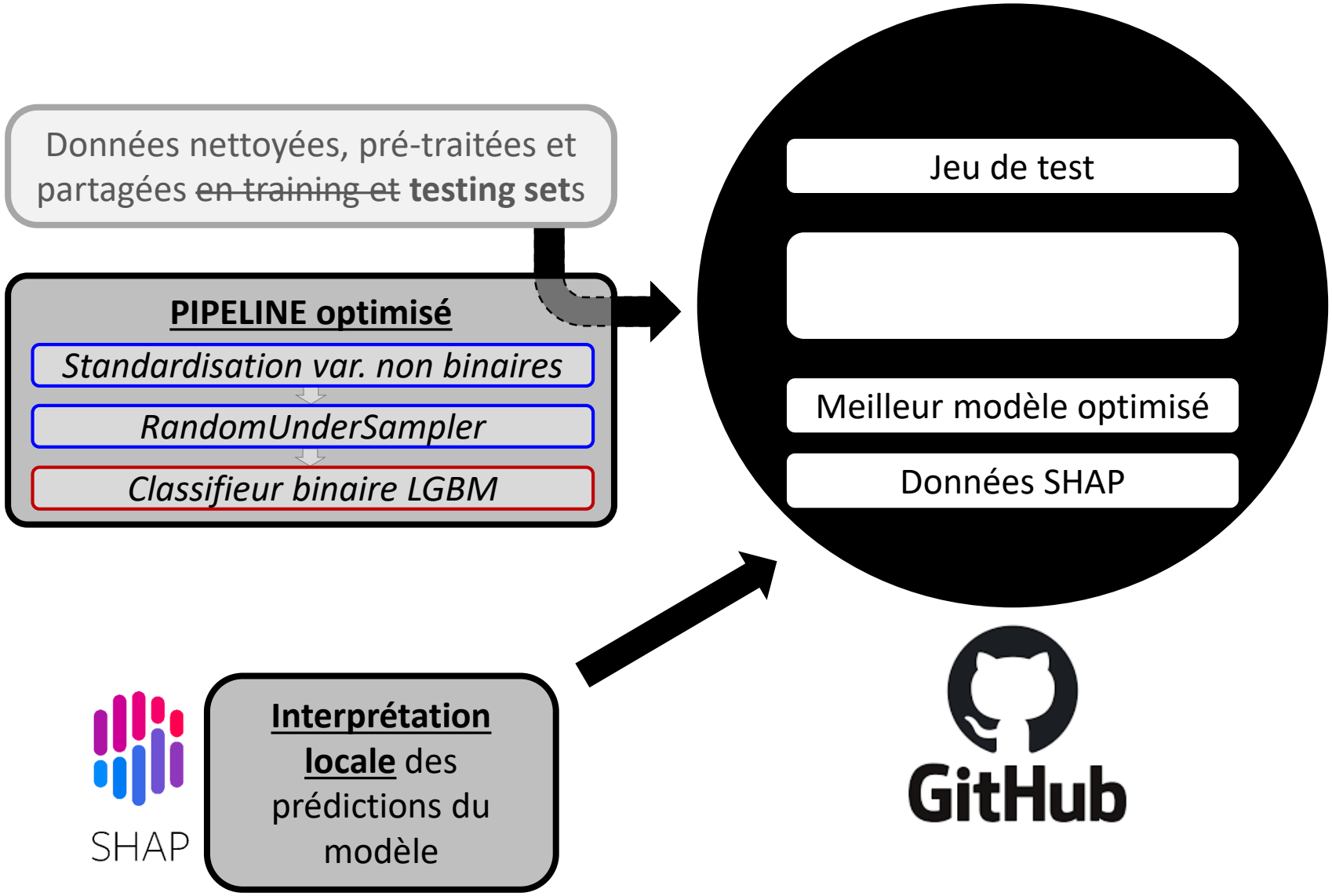
```
{"predicted probability":0.7850252299604359,"prediction":"Sorry, loan ungranted..."}
```

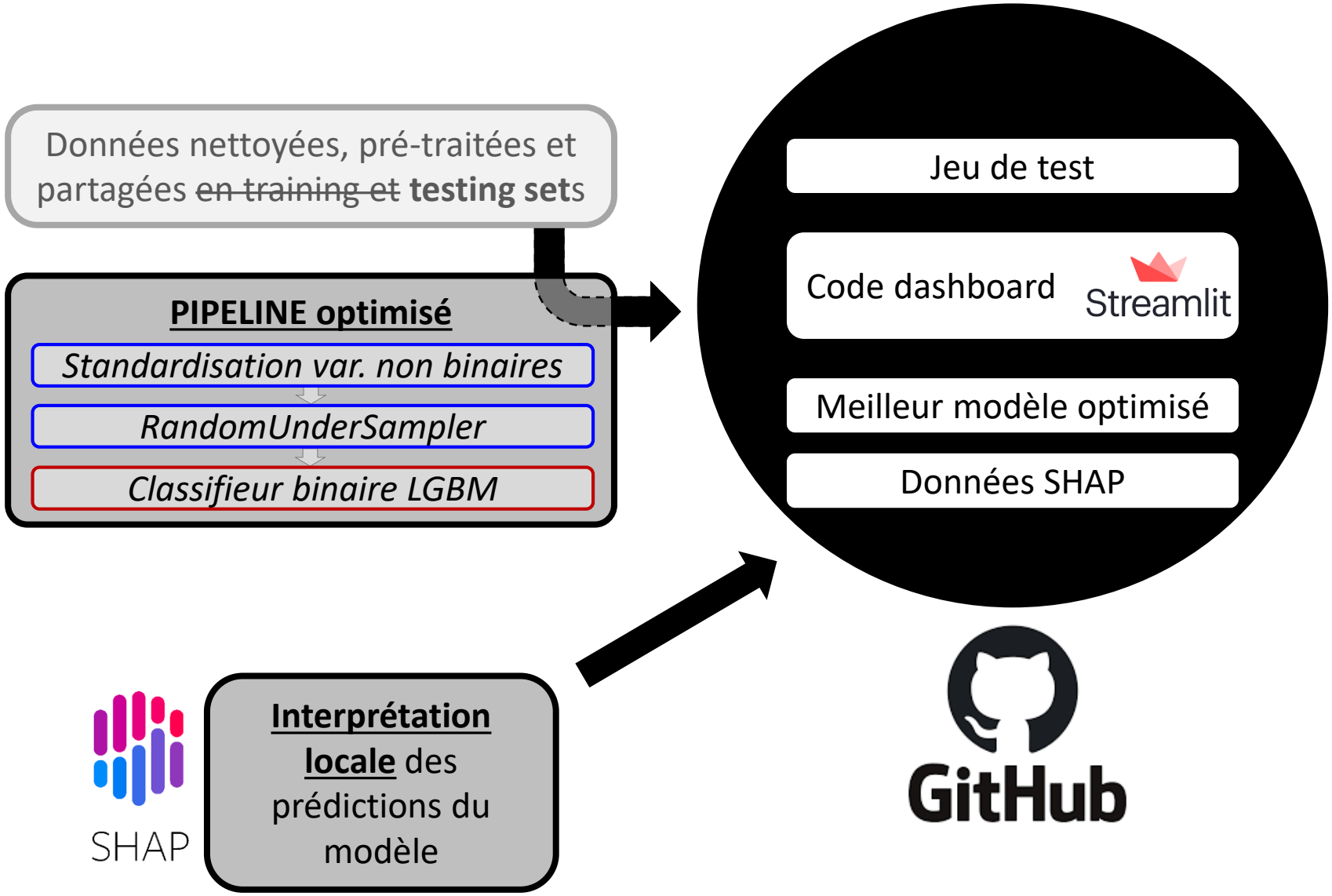
Déploiement du modèle sous forme d'API

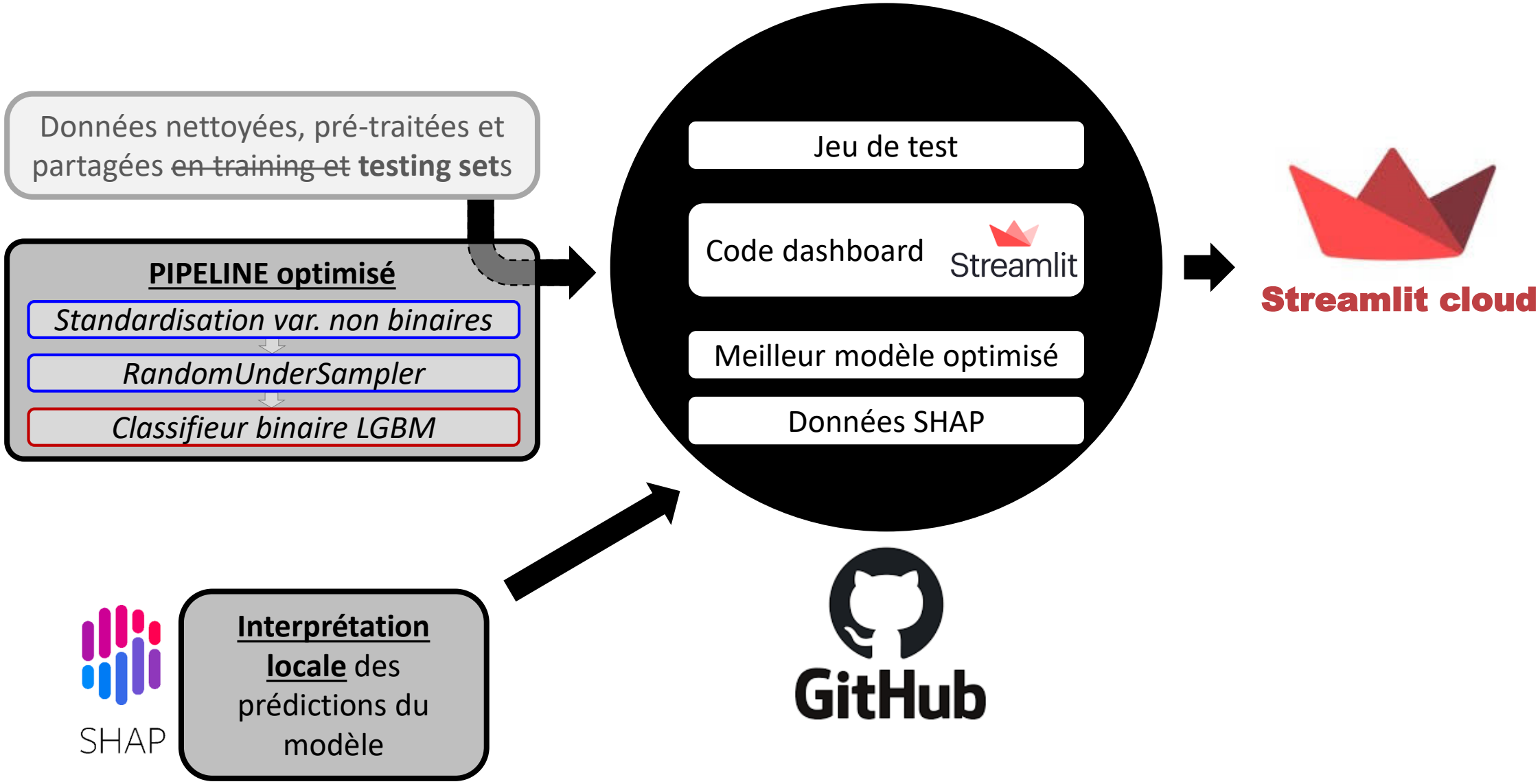
FastAPI

heroku


Données nettoyées, pré-traitées et partagées ~~en training et~~ **testing set**s
PIPELINE optimisé
Standardisation var. non binaires
RandomUnderSampler
Classifieur binaire LGBM
Jeu de test
Meilleur modèle optimisé
GitHub


Données nettoyées, pré-traitées et partagées ~~en training et~~ **testing set**s
PIPELINE optimisé
Standardisation var. non binaires
RandomUnderSampler
Classifieur binaire LGBM
Jeu de test
Meilleur modèle optimisé
Données SHAP
SHAP
Interprétation locale des prédictions du modèle
GitHub


Données nettoyées, pré-traitées et partagées ~~en training et~~ **testing sets**
PIPELINE optimisé
Standardisation var. non binaires
RandomUnderSampler
Classifieur binaire LGBM
Jeu de test
Code dashboard
Streamlit
Meilleur modèle optimisé
Données SHAP
SHAP
Interprétation locale des prédictions du modèle
GitHub


Données nettoyées, pré-traitées et partagées ~~en training et~~ **testing sets**
PIPELINE optimisé
Standardisation var. non binaires
RandomUnderSampler
Classifieur binaire LGBM
Jeu de test
Code dashboard
Streamlit
Meilleur modèle optimisé
Données SHAP
Streamlit cloud
GitHub
SHAP
Interprétation locale des prédictions du modèle

MISSIONS

☐ Construire un **modèle** de scoring qui donnera une prédiction sur la probabilité de faillite d'un client de façon automatique.

☐ Construire un **dashboard** interactif à destination des gestionnaires de la relation client permettant d'interpréter les prédictions faites par le modèle

MISSIONS

☑ Construire un **modèle** de scoring qui donnera une prédiction sur la probabilité de faillite d'un client de façon automatique.

☐ Construire un **dashboard** interactif à destination des gestionnaires de la relation client permettant d'interpréter les prédictions faites par le modèle

MISSIONS

☑ Construire un **modèle** de scoring qui donnera une prédiction sur la probabilité de faillite d'un client de façon automatique.

☐ Construire un **dashboard** interactif à destination des gestionnaires de la relation client permettant d'interpréter les prédictions faites par le modèle

Améliorations possibles à tester :

- Combinaison SMOTE puis RandomUnderSampling
- Optimisation simultanée de l'ensemble des hyperparamètres + sur une durée plus longue
- Ré-entrainement du modèle sur l'ensemble du jeu de données (train + test)
- Optimisation du seuil avec une fonction de coût permettant d'optimiser non pas simplement la détection des mauvais payeurs mais les profits réalisés par la banque
- Suppression de la variable reflétant le genre du client

MISSIONS

☑ Construire un **modèle** de scoring qui donnera une prédiction sur la probabilité de faillite d'un client de façon automatique.

☑ Construire un **dashboard** interactif à destination des gestionnaires de la relation client permettant d'interpréter les prédictions faites par le modèle

Améliorations possibles à tester :

- Combinaison SMOTE puis RandomUnderSampling
- Optimisation simultanée de l'ensemble des hyperparamètres + sur une durée plus longue
- Ré-entrainement du modèle sur l'ensemble du jeu de données (train + test)
- Optimisation du seuil avec une fonction de coût permettant d'optimiser non pas simplement la détection des mauvais payeurs mais les profits réalisés par la banque
- Suppression de la variable reflétant le genre du client

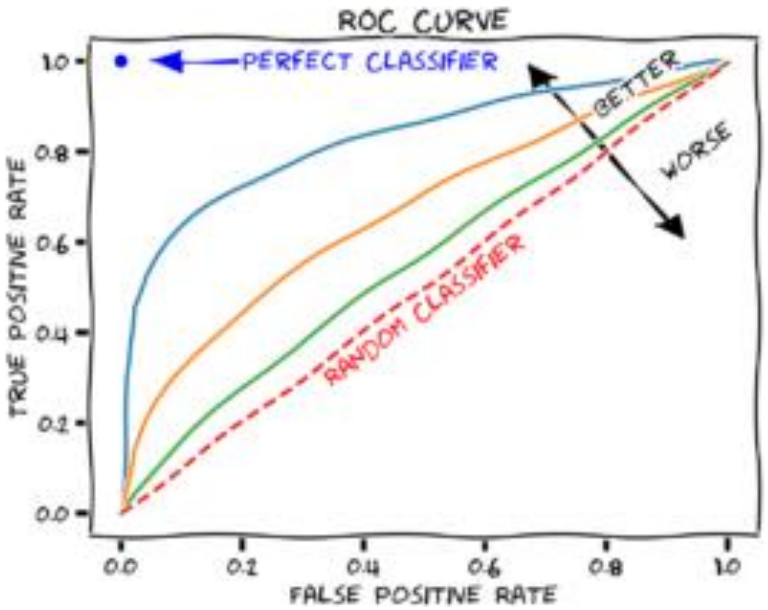
Changements qui devraient être apportés en conditions réelles :

- Améliorations du modèle
- Jeu de données inconnu réel (e.g. 'application_test.csv') + adaptations nécessaires associées (e.g. recalcul SHAP)
- 'Désencodage' des variables catégorielles
- Noms plus explicites des variables

ROC-AUC ➔ adapté pour classes équilibrées

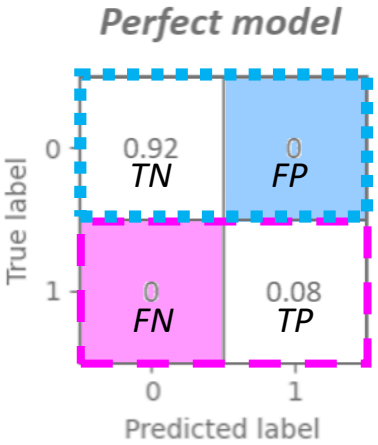
Rappel (Sensibilité):

$$\frac{TP}{TP + FN}$$



1 – Spécificité:

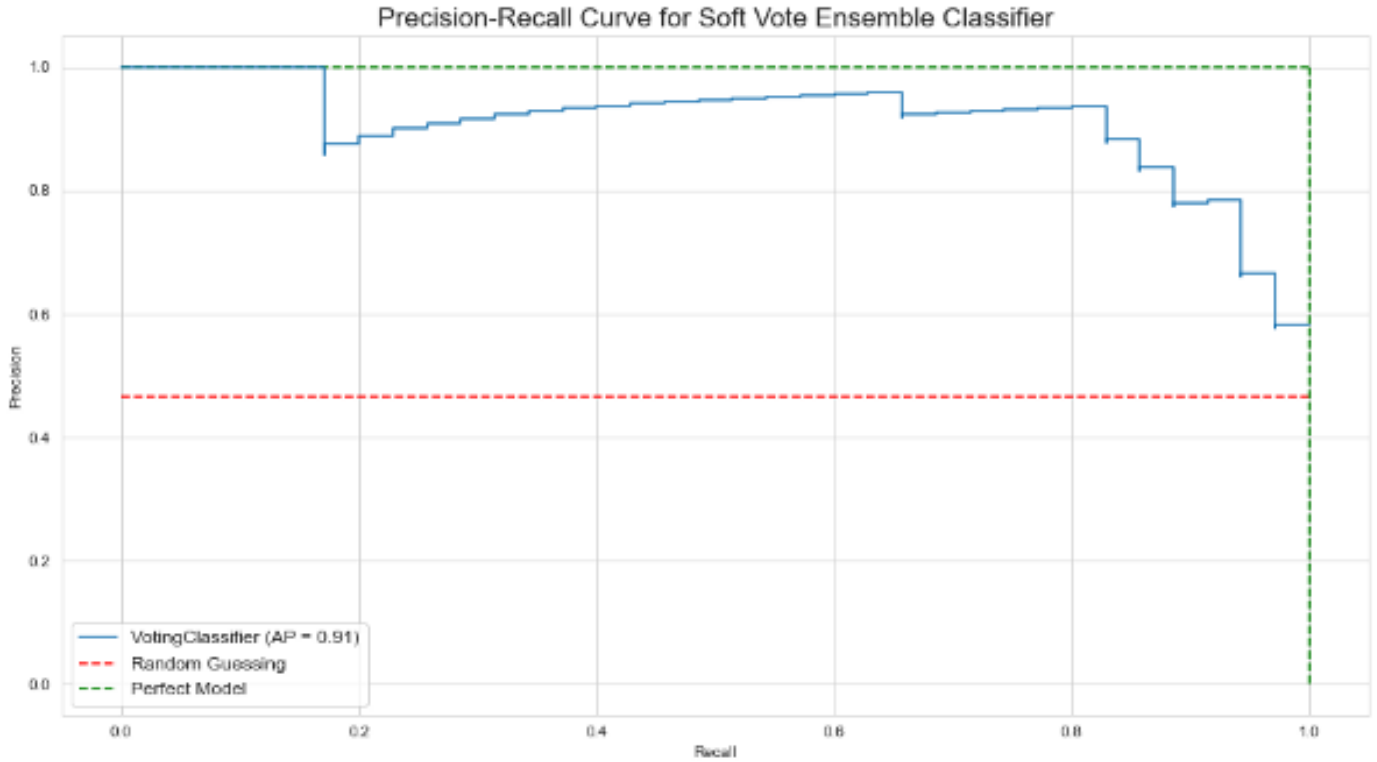
$$1 - \frac{TN}{FP + TN}$$



https://www.idbc.fr/tutoriel-comment-lire-une-courbe-roc-et-interpreter-son-auc/

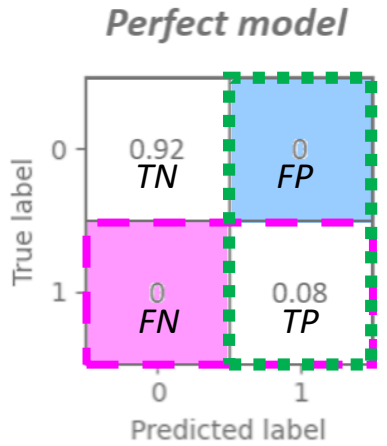
PR curve ➔ adaptée pour classes deséquilibrées

Only concerned about the skill to predict the minority ('1') class / doesn't make use of the TN (i.e. skill of the model at predicting the majority ('0') class correctly, i.e. high TN).



Précision:

$$\frac{TP}{TP + FP}$$



Rappel (Sensibilité):

$$\frac{TP}{TP + FN}$$

Sklearn « average_precision_score » metrics:

summarizes a precision-recall curve (as the weighted mean of precisions achieved at each threshold, with the increase in recall from the previous threshold used as the weight)…

/!\ different from computing the area under the precision-recall curve

https://ichi.pro/fr/courbes-de-rappel-de-precision-pour-les-ensembles-de-donnees-desequilibres-et-lies-a-la-sante-250398578793758

F-score versus area under ROC or PR curve

F-score summarizes model skill for a specific probability threshold (e.g. 0.5)

whereas the area under curve summarize the skill of a model across thresholds

Approche générale

Données nettoyées, pré-traitées et partagées en training et testing sets

PIPELINE

- *Imputation NaN par la médiane*
- *Standardisation des variables numériques non binaires*
- *Sur/sous-échantillonnage des classes minoritaires/majoritaires*
- *Classifieur binaire*

Test / sélection des méthodes de pré-traitements et modèles de classification binaire supervisée

Optimisation des hyperparamètres des méthode(s) et modèle les plus performants

Optimisation du seuil de probabilité conditionnant l'attribution des classes

Interprétation globale et locale des prédictions du modèle

Déploiement du modèle sous forme d'API