

# P7 – Implémenter un modèle de scoring

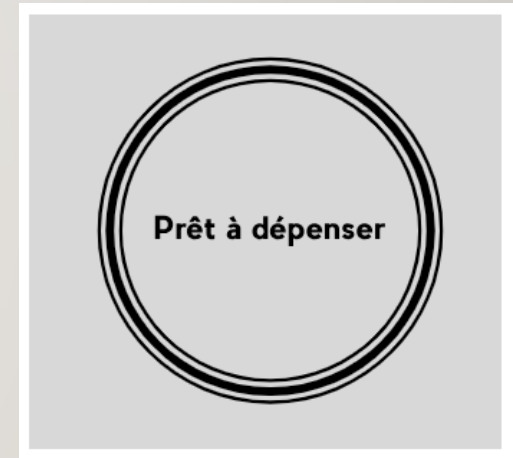
---

MARWA EL HOURI

# Problématique

---

- Prêt à dépenser est une société financière qui propose des crédits à la consommation pour des personnes ayant peu ou pas du tout d'historique de prêts
- Notre objectif:
  - Mettre en place un outil de « scoring » pour calculer la probabilité qu'un client rembourse son crédit.
  - Développer un Dashboard interactif pour que les chargés de relation client puissent expliquer clairement les raisons de leur décision.



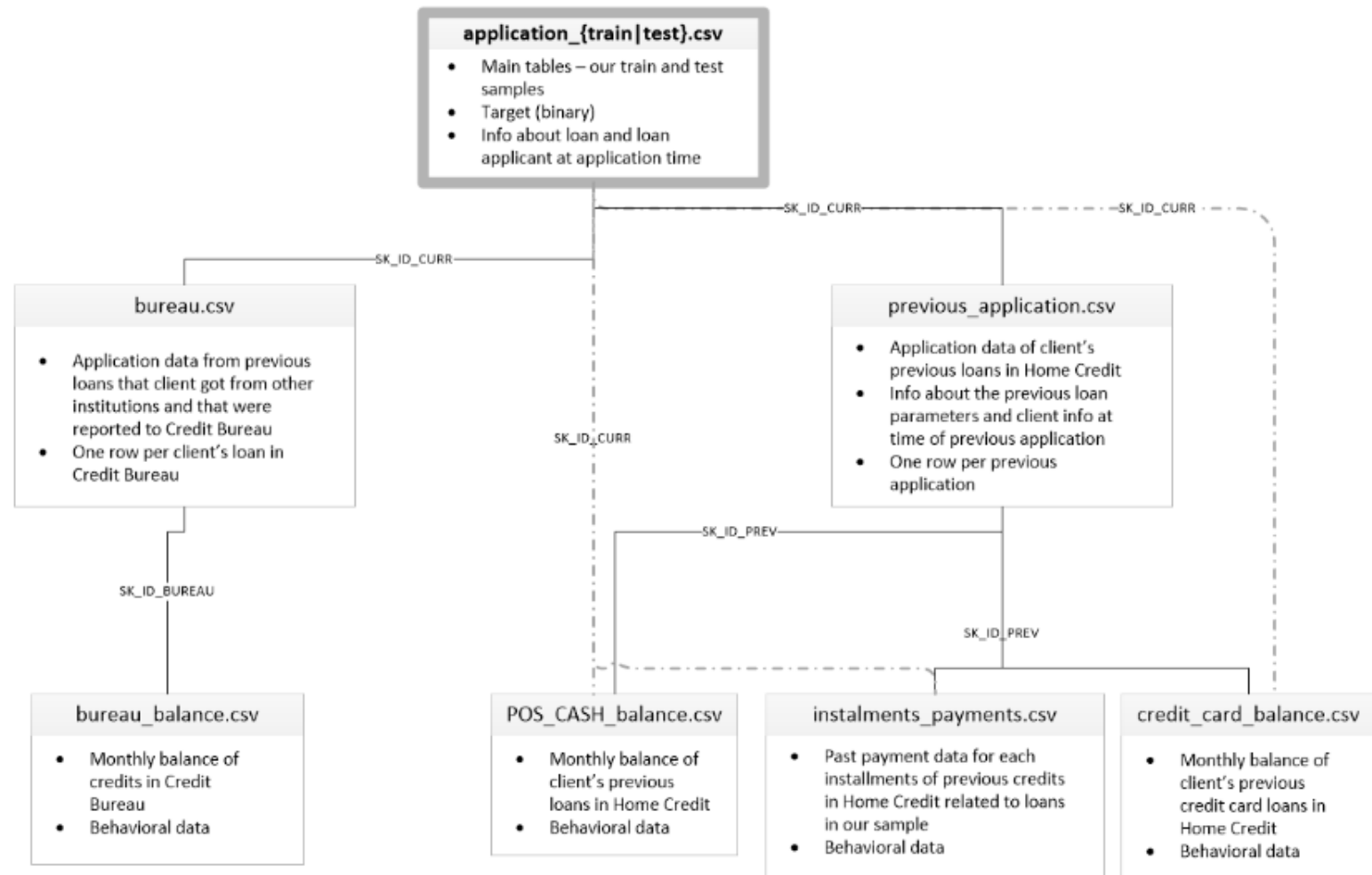
# Plan

---

1. Préparation et nettoyage des données
2. Recherche du modèle
3. Présentation du Dashboard interactif
4. Conclusion

# I- Préparation et nettoyage des données

7 jeux de données  
contenant des  
informations  
personnelles et  
financière des clients



# I- Préparation et nettoyage des données

---

- Merge et feature engenning en utilisant le kernel kaggle

<https://www.kaggle.com/jsaguiar/lightgbm-with-simple-features>

- Encoder les variables catégorielles
- Créer des agrégations des variables en calculant des grandeurs statistiques (min, max, mean, var, sum)
- Joindre les différents jeux de données

Résultat: un jeu de données de dimensions (307 507, 797)

- Nettoyage des données:

- Enlever les variables a plus de 50% de valeurs manquantes cela permettra d'enlever 236 variables
- Enlever les variables à variance nulle (19 variables)
- Enlever les variables à 95% nulle (98 variables)
- Remplacer les variables manquantes quantitative par la médiane.

Résultat: un jeu de données de dimensions (307 507, 444)

## 2- Recherche du modèle

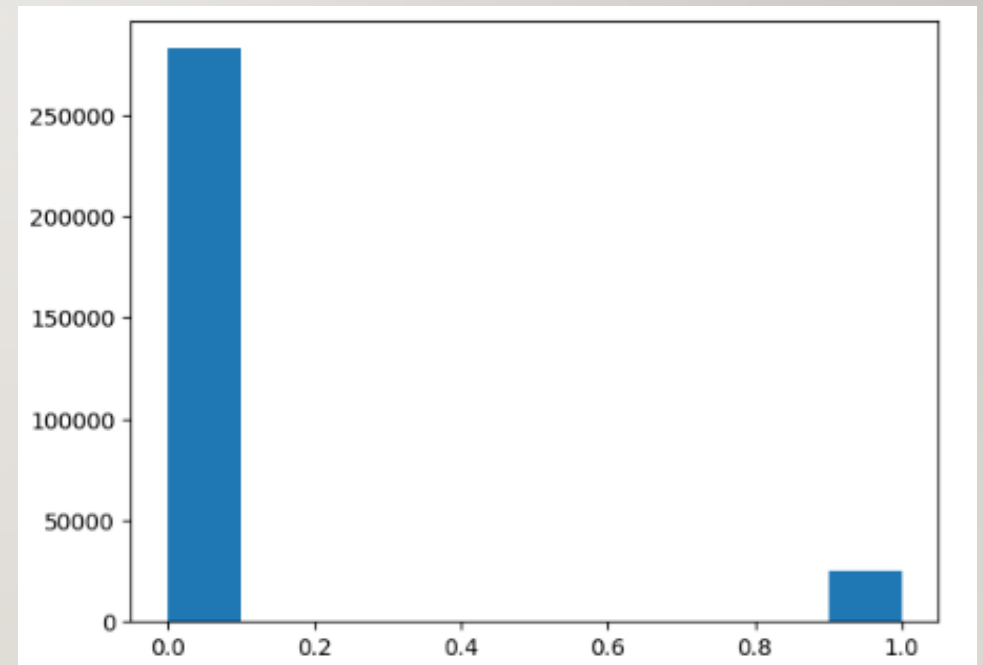
---

- Problème des données déséquilibrées:
- Score personnalisée
- Recherche des hyperparamètres
- Résultats

## 2.1- Problème des données déséquilibrées:

---

- Présences de 2 classes « TARGET » :
  - Classe 0 : éligible au crédit : 92% des instances
  - Classe 1 : non éligible au crédit : 8% des instances
- Le rapport des classes est de l'ordre de 11



## 2.1 - Problème des données déséquilibrées:

---

- Utilisation du paramètre « class-weight » présent dans les méthodes de classification ensemblistes
- Utilisation de SMOTE pour rééquilibrer les classes
- Utilisation des classificateurs de la librairie « imblearn » tel « BalancedRandomForest » prédéfini à prendre en compte des jeux de données déséquilibrées.



## 2.2- Score personnalisé

---

- Attribuer un crédit à un client non éligible est beaucoup plus coûteux que refuser un crédit à un client éligible.
- Objectif:
  - Minimiser la mauvaise classification des classes (Faux négatifs et faux positifs)
  - Minimiser principalement les faux négatifs (donner un crédit aux clients non éligible) ,
- Solution:
  - Construite un score personnalisé en donnant plus de poids aux faux négatifs qu'aux faux positifs.
  - On choisit donc d'utiliser la fonction de score pour la mesure de performance du modèle :

$$\text{custom\_metric} = 11 FN + FP$$

## 2.3 Recherche des hyperparamètres

---

### **Classifieur à tester:**

- Balanced Random Forest
- Random Forest Classifier
- LightGBM

Pour chaque  
classificateur:

Recherche des  
hyperparamètres  
par hyperopt

Récupérer  
le meilleur  
modèle

Application de  
l'algorithme  
d'optimisation

Récupérer  
score et seuil  
de décision

## 2.3 Recherche des hyperparamètres

---

- Pour la recherche d'hyperparamètres on utilise la recherche Bayésienne de la librairie « HyperOpt »
- On définit une fonction de minimisation de score qui utilise
  - la cross validation avec « stratifiedKfold » sur 3 partitions
  - et la méthode de « scoring » avec la fonction de score personnalisée
- Pour le modèle avec les meilleurs paramètres est trouvé, on effectue une étape d'optimisation supplémentaire
  - Recherche récursive du seuil de décision qui minimise la fonction de score personnalisée.

## 2.3 - Résultats

---

Modèle	Score	Score avant optimisation	Score après optimisation	Seuil
Balanced Random Forest Classifier	39950	58936	58863	0.5050
Random Forest Classifier	40894	60099	6001	0.4949
Lightgbm	<b>36430</b>	<b>53523</b>	<b>53 454</b>	<b>0.4747</b>

## 2.3 - Résultats

---

- Modèle à utiliser:
  - Lightgbm
  - sans oversampling
  - avec class\_weight=balanced

- Matrice de confusion du modèle

	0	1
0	65814	27494
1	2360	5810

# 3- Présentation du Dashboard interactif

---

- Outils pour la conception et déploiement du Dashboard
- Conception de l'application
- Présentation du Dashboard interactif

## 3.1 Outils pour la conception et le déploiement du tableau de bord

---

- Création du tableau de bord :



- RestFul API :



- Déploiement Cloud :



- Repository (dépôt du code) :



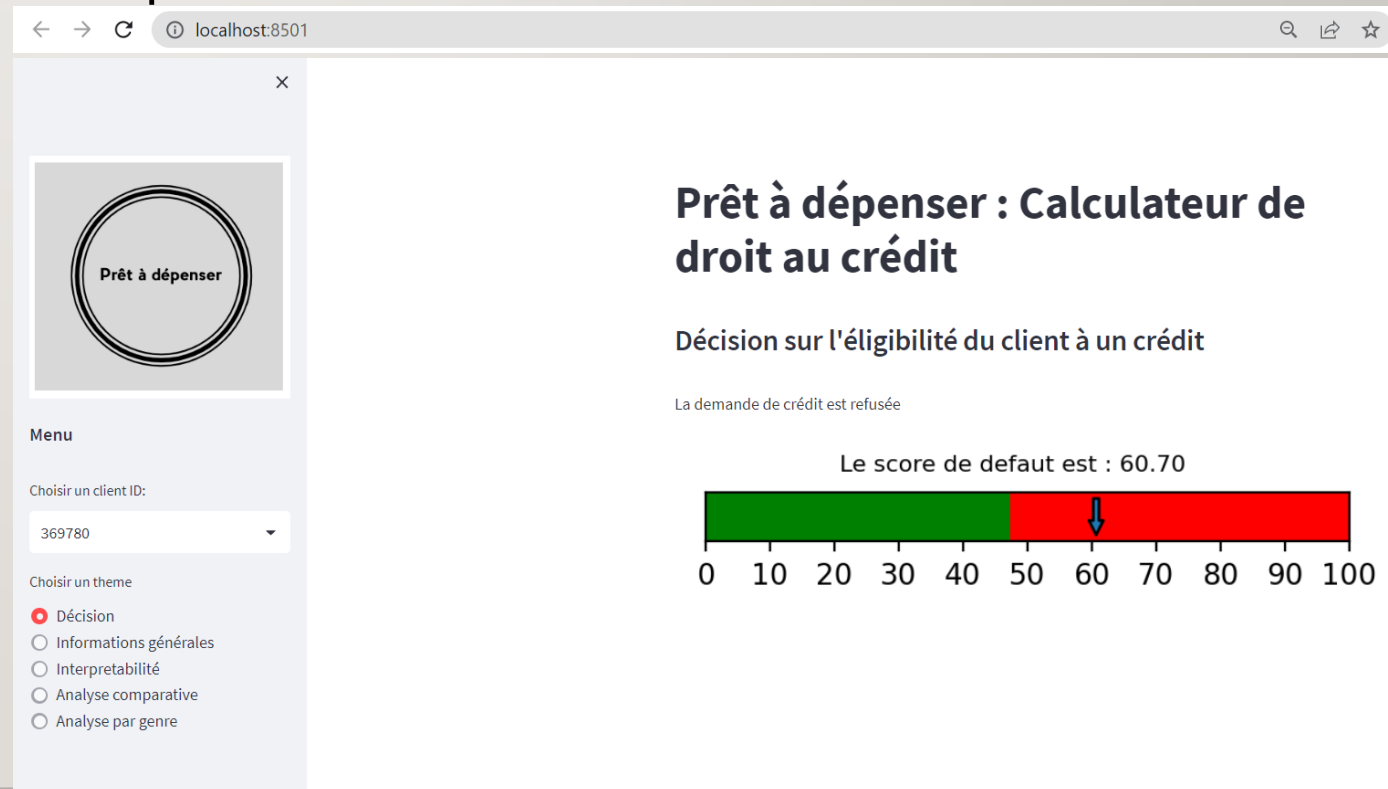
# Création du tableau de bord



- Streamlit : Ensembles d'outils pour la conception et création de tableau de bord en utilisant Python

- URL d'accès :

<http://52.47.203.229:8501>





# RestFul API :



- L'accès aux information dans les jeux de données est centralisé dans l'API Flask
- URL d'accès :









<http://52.47.203.229:5000>

```
← → ↻ ⓘ 127.0.0.1:5000
[
  {
    "doc": "Main page - Prints endpoints and their documentation",
    "endpoint": "/"
  },
  {
    "doc": " Returns the list of client ids",
    "endpoint": "/clients"
  },
  {
    "doc": " Returns the information of a given client from the complete dataset",
    "endpoint": "/clients-info/<int:id>"
  },
  {
    "doc": " Returns the prediction using personalised threshold, the threshold and the prediction probabilities for a given client",
    "endpoint": "/clients/<int:id>"
  },
  {
    "doc": " Returns for the given feature the distribution of the feature (min, 25%, median, 75%, max), list of client ids having the minimum value and list of client ids having the maximum value for the feature, feature examples: PAYMENT_RATE, EXT_SOURCE_2,EXT_SOURCE_3, DAYS_BIRTH... ",
    "endpoint": "/feature-info/<feature>"
  },
  {
    "doc": " Displays boxplots by gender for the best nine indicators along with the position of the client with respect to the dataset",
    "endpoint": "/gender/<int:id>"
  },
  {
    "doc": " Display the kde graph for the actual classes of clients for a given feature ",
    "endpoint": "/kde/<int:id>/<feature>"
  },
  {
    "doc": " Returns shap informations of a given client ",
    "endpoint": "/shap/<int:id>"
  },
]
```

# Repository (dépôt du code) :



- Tous les codes et ressources de déploiement sont disponible sur :
- URL d'accès : <https://github.com/MarwaHouri/Open-Classroom-Projet-7-Streamlit>

 MarwaHouri add getData.py and flask_routes.txt			bec43c8 5 days ago	 29 commits
 P7-Flask.py	Change in folder architecture			5 days ago
 P7-Streamlit.py	Change in folder architecture			5 days ago
 README.md	Update README.md			last week
 flask_routes.txt	add getData.py and flask_routes.txt			5 days ago
 getData.py	add getData.py and flask_routes.txt			5 days ago
 requirements.txt	Change in folder architecture			5 days ago

# Déploiement Cloud :



- Le déploiement de l'API et du Dashboard interactif est fait sur AWS EC2

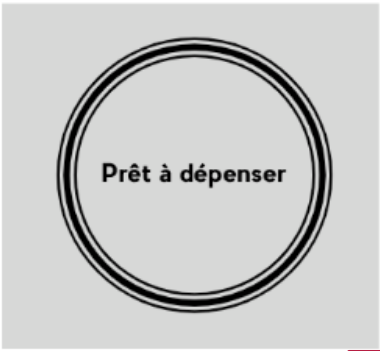
```
admin@ip-172-31-39-180: ~/Open-Classroom-Projet-7-Streamlit
151 git clone https://github.com/MarwaHouri/Open-Classroom-Projet-7-Streamlit.git
152 ls -l
153 cd Open-Classroom-Projet-7-Streamlit/
154 ls -l
155 cat README.md
156 pip install -r requirements.txt
157 python getData.py
158 ls -l
159 ls -l Ressources/
160 ls -l Ressources/datasets/
161 nohup flask --app P7-Flask.py run --host=0.0.0.0 &
162 cat nohup.out
163 nohup streamlit run P7-Streamlit.py &
164 cat nohup.out
165 history
(p7) admin@ip-172-31-39-180:~/Open-Classroom-Projet-7-Streamlit$
```

## 3.2 Conception de l'application

---

- **Décision** : Visualisation de la décision
- **Informations générales** : Informations générales sur le model de décision
- **Interprétabilité** : Interprétabilité des résultats du client
- **Analyse comparative** : Situation du client vis-à-vis des autres
- **Analyse par genre** : Distribution des indicateurs par genres et situation du client par rapport a ces distributions

# Calculateur de droit au crédit – Page d'accueil



Prêt à dépenser

Menu

Choisir un client ID:

369780

Choisir un theme

- ☒ Décision
- ☐ Informations générales
- ☐ Interprétabilité
- ☐ Analyse comparative
- ☐ Analyse par genre

Liste déroulante des client ID

Radio buttons pour le choix des informations à visualiser

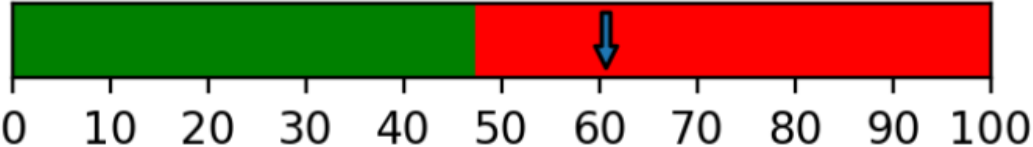
## Prêt à dépenser : Calculateur de droit au crédit

Décision sur l'eligibilité du client à un crédit

La demande de crédit est refusée

Décision du modèle

Le score de défaut est : 60.70



Représentation graphique du score

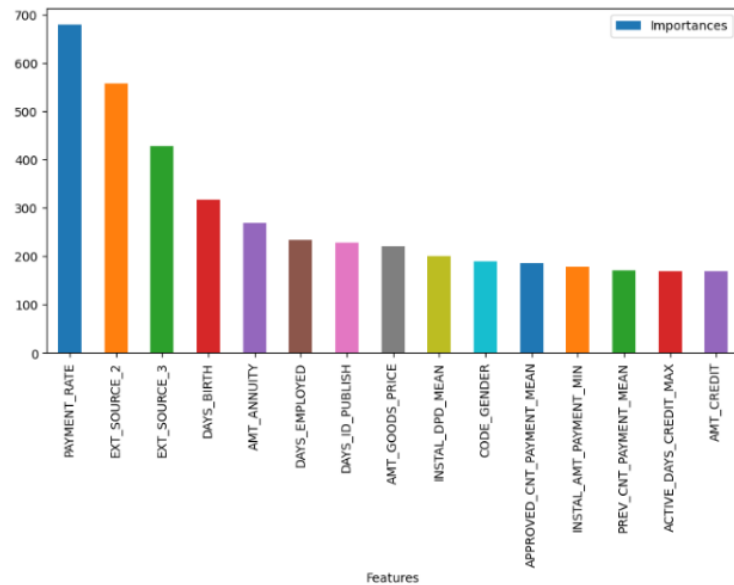
# Calculateur de droit au crédit

## Informations générales sur le modèle

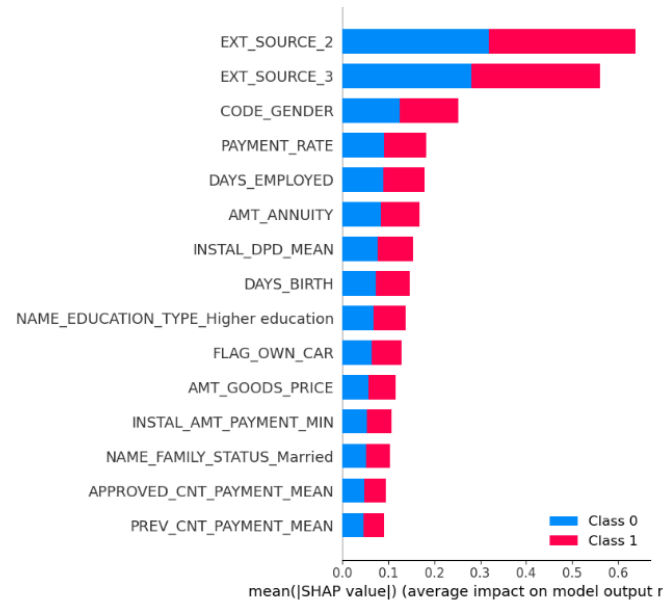
Choisir un theme

- ☐ Décision
- ☒ Informations générales
- ☐ Interprétabilité
- ☐ Analyse comparative
- ☐ Analyse par genre

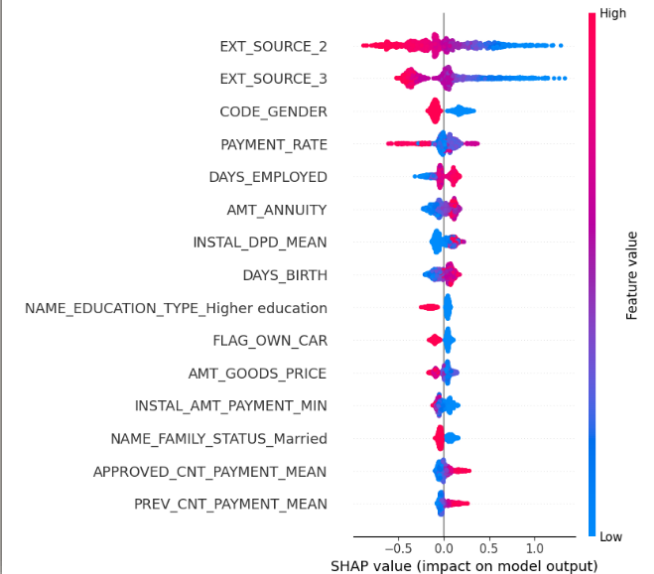
Top 15 features importances générées par le modèle



Impact moyen des indicateurs sur la décision (SHAP)



Shap summary plot : impact des indicateurs sur la prédiction de rejet par instance:



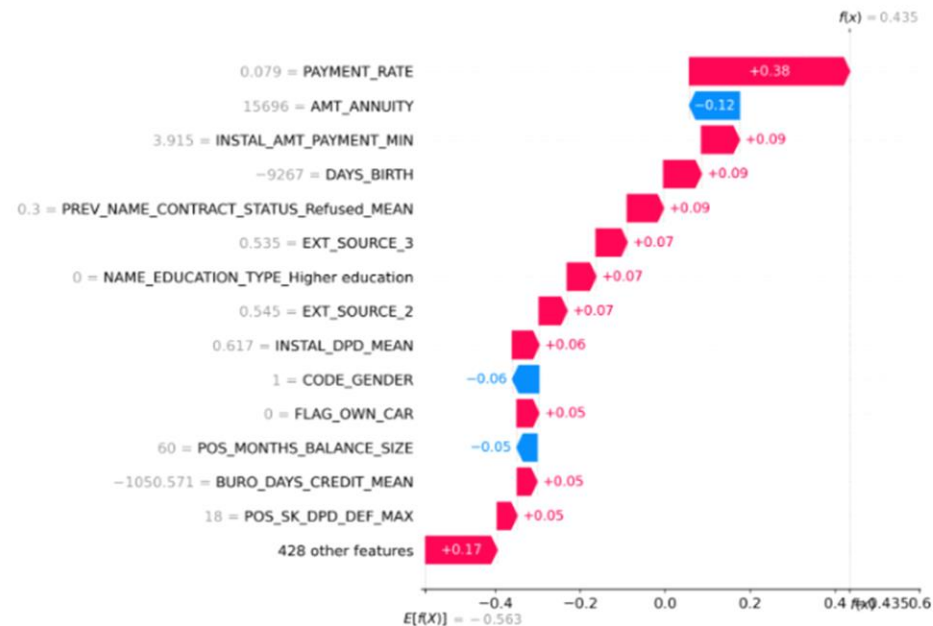
# Calculateur de droit au crédit

## Interprétation locale de la décision

Choisir un theme

- ☐ Décision
- ☐ Informations générales
- ☒ Interprétabilité
- ☐ Analyse comparative
- ☐ Analyse par genre

Shap waterfall



Shap force plot du client





# Calculateur de droit au crédit

## Analyse comparative du client par rapport aux autres clients

Choisir un theme

- ☐ Décision
- ☐ Informations générales
- ☐ Interprétabilité
- ☒ Analyse comparative
- ☐ Analyse par genre

Choisir les indicateurs :

PAYMENT\_RATE x EXT\_SOURCE\_2 x

EXT\_SOURCE\_3

DAYS\_BIRTH

AMT\_ANNUITY

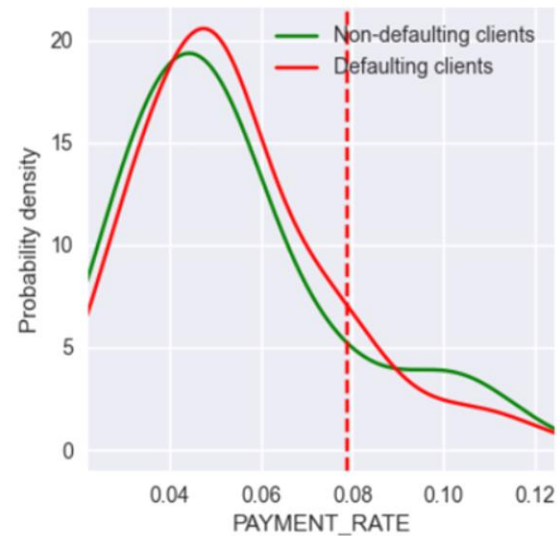
DAYS\_EMPLOYED

DAYS\_ID\_PUBLISH

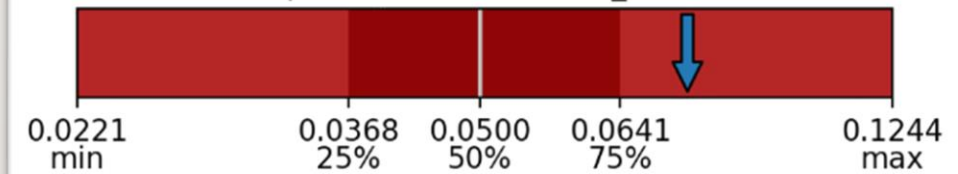
AMT\_GOODS\_PRICE

INSTAL\_DPD\_MEAN

Distribution de PAYMENT\_RATE par rapport a la vrai classe des cli



Valeur client pour feature PAYMENT\_RATE est 0.079



☒ Montrer la liste des clients ayant une valeur minimale

☐ Montrer la liste des clients ayant une valeur maximale

Client ID
0 334592



# Calculateur de droit au crédit

## Exploration des indicateurs en fonction du genre du client

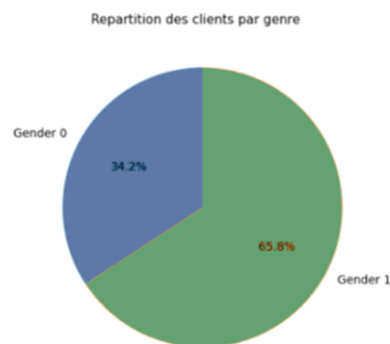
Choisir un theme

- ☐ Décision
- ☐ Informations générales
- ☐ Interprétabilité
- ☐ Analyse comparative
- ☒ Analyse par genre

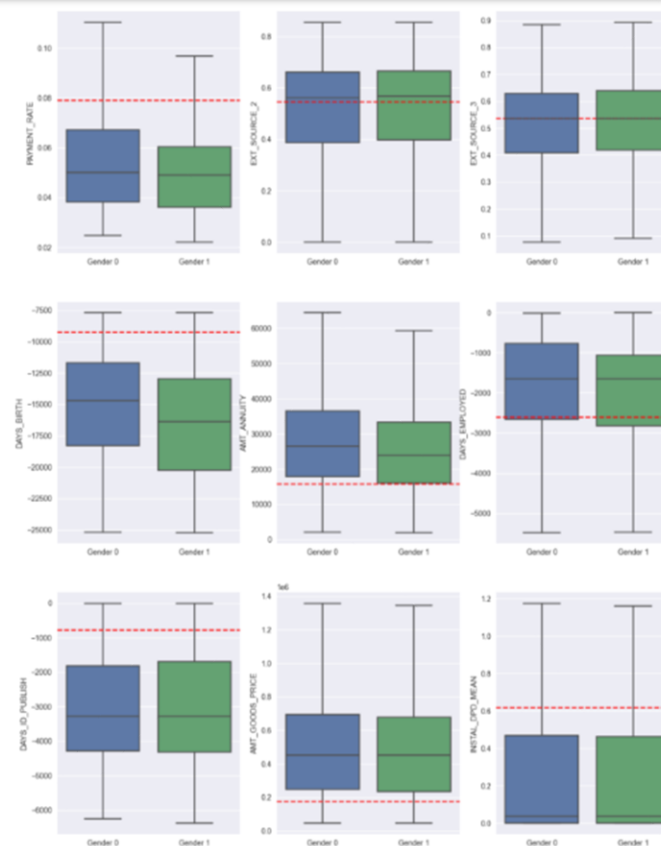
### Informations client

	Informations client 369780
PAYMENT_RATE	0.0790
EXT_SOURCE_2	0.5449
EXT_SOURCE_3	0.5353
DAYS_BIRTH	-9,267.0000
AMT_ANNUITY	15,696.0000
DAYS_EMPLOYED	-2,602.0000
DAYS_ID_PUBLISH	-785.0000
AMT_GOODS_PRICE	175,500.0000
INSTAL_DPD_MEAN	0.6167
CODE_GENDER	1.0000

### Distribution des clients par genre



### Situation du client par rapport aux indicateurs principaux en fonction du genre



# Conclusion : Limitations et Améliorations

---

- Préparation des données
  - Une discussion avec des experts du métier permettra d'améliorer le « feature engineering »
- Recherche du meilleur modèle
  - Explorer d'autres moteurs de classification
  - Augmenter le nombre de paramètres à tester
  - Tester d'autres méthodes pour rééquilibrer le jeu de données
- Fonction de score personnalisée
  - Vision experte du coût réel de la mauvaise classification et des résultats de prédiction acceptables

---

Merci!