Projet 7: Implémentez un modèle de scoring

QUENTIN STEPNIEWSKI

OPENCLASSROOMS

23 NOV 2020

Sommaire

1. Introduction – Présentation de la problématique

Data Scientist au sein d'une société financière nommée « prêt à dépenser », qui propose des crédits à la consommation pour des personnes avec peu d'historique de prêt

Problématique principale

- Développer un **modèle de scoring** de la **probabilité de défaut de paiement** d'un client

Objectifs de l'étude

- **Construction d'un modèle de scoring** adapté
- **Mise en place d'un dashboard interactif** à destination des gestionnaires de la relation client

2. Présentation des données utilisées

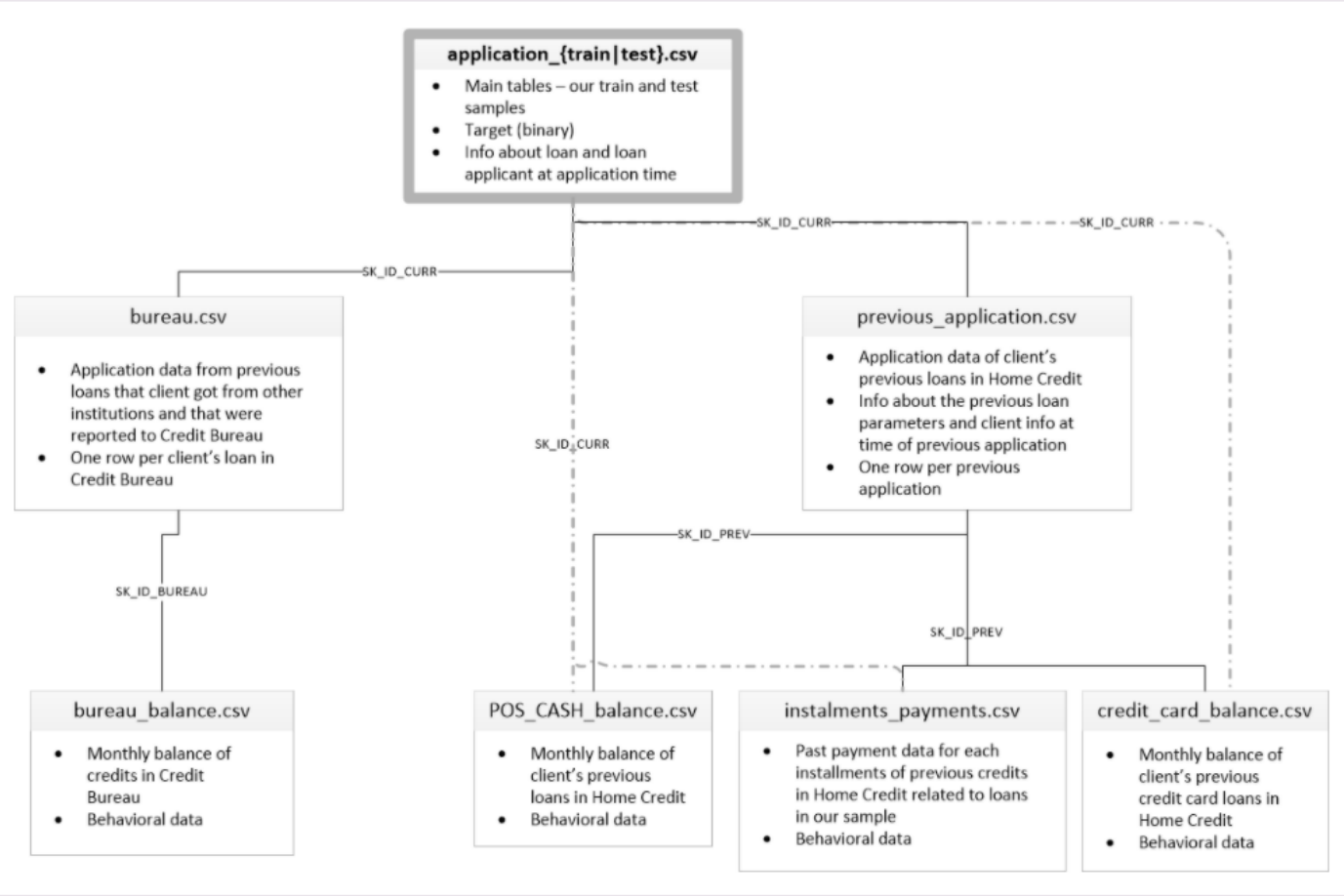
Jeu de données provenant de 7 sources différentes :

Informations principales sur la base de données :

- Plus de 300 000 clients

- 120 features (âge, sexe, emploi, logement, revenus, informations relatives au crédit…)

- Feature cible :
  - Défaut de crédit (catégorie 1)
  - Pas de défaut de crédit (catégorie 0)



2. Présentation des données utilisées

Preprocessing :

Utilisation d'un preprocessing existant : Notebook Kaggle

Le notebook met en plus plusieurs étapes de preprocessing:

- One hot encoding de variables catégorielles

- Détection d'outliers/valeurs aberrantes

- Création de features spécifiques à la problématique :
  - Ratio du montant du crédit par rapport au revenu du client
  - Ratio des annuités par rapport au revenu du client
  - Durée du prêt en mois
  - Pourcentage de jours salariés par rapport à l'âge du client

- Imputation de valeurs manquantes (SimpleImputer median)

=> 240 Features en sortie, exploitables pour notre modélisation

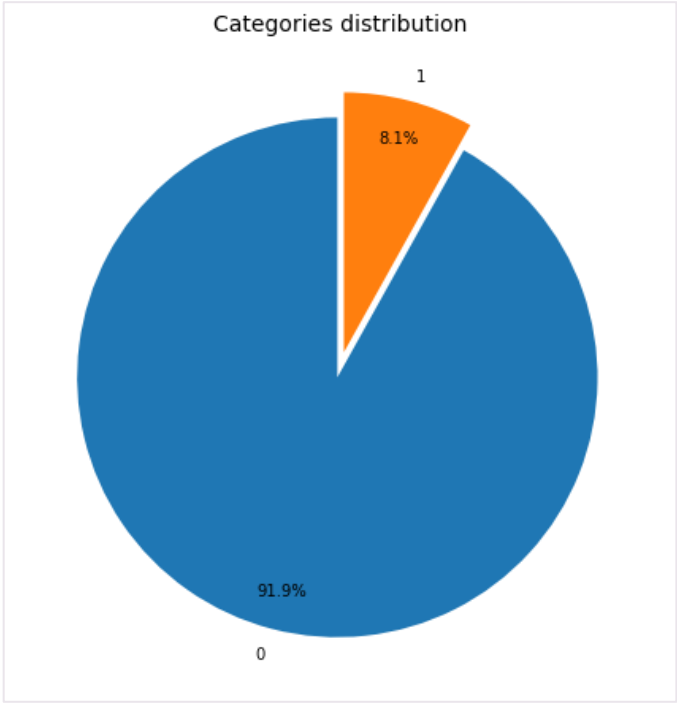
2. Présentation des données utilisées

Problématique principale des données :

Très large déséquilibre entre les classes :

- Clients sans défaut (classe 0) : **92%** du dataset

- Clients avec défaut (classe 1) : **8%** du dataset



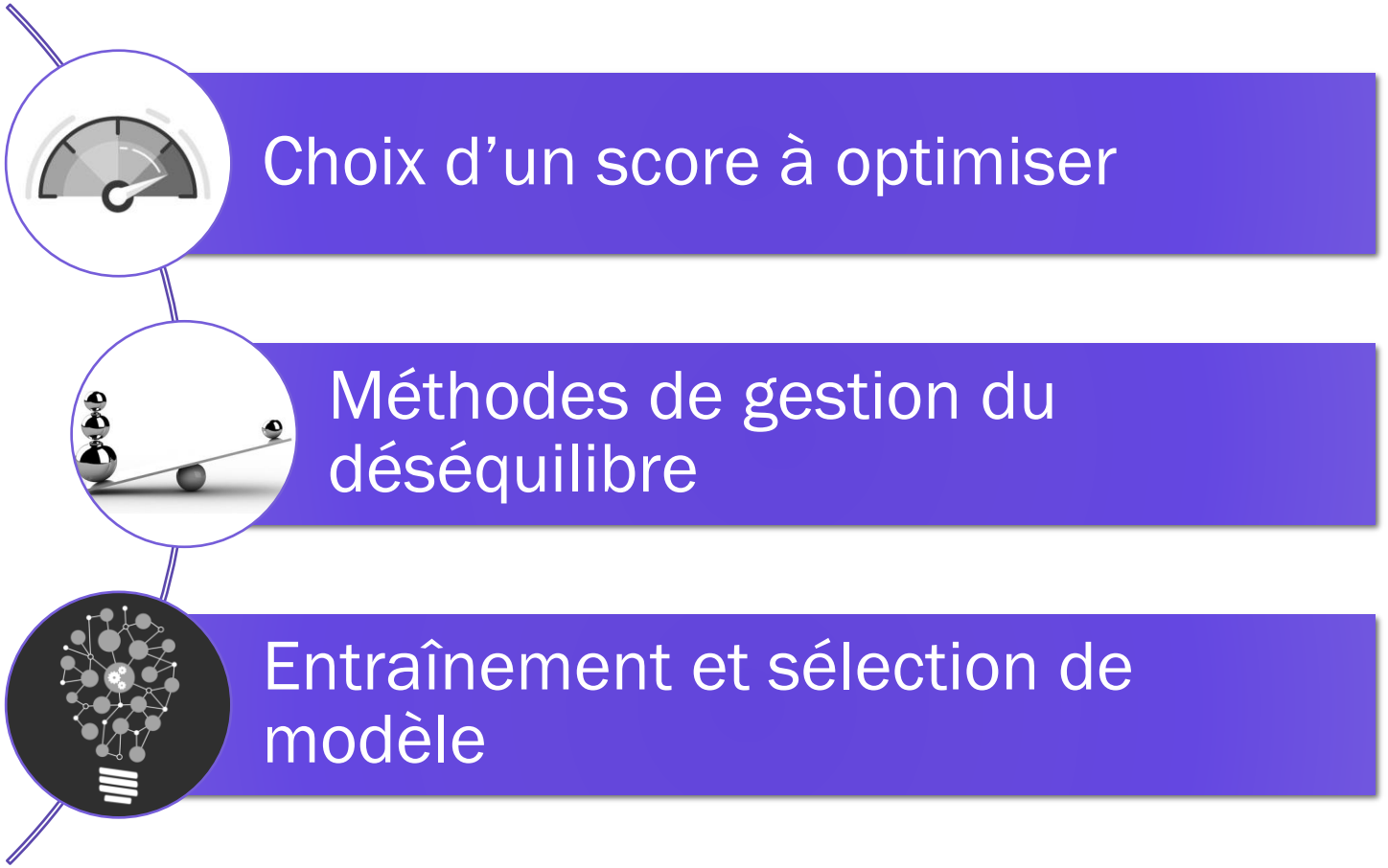
Déséquilibre problématique pour l'apprentissage du modèle :

➢ *Risque de prédire uniquement la classe majoritaire (**accuracy de 92%**)*

Enjeu principal du projet : gestion de ce déséquilibre

3. Approche de modélisation

3 axes principaux :



3. Approche de modélisation

Choix d'un score pertinent :

2 types d'erreurs possibles :

- Clients à risques non identifiés (pertes/somme non-recouvrées) : **Faux négatif**

- Clients peu risqués et identifiés comme risqués (coût d'opportunité) : **Faux positif**

Faux négatif plus risqué que faux positif

	Prédit sans défaut (0)	Prédit en défaut (1)
Réel sans défaut (0)	Vrai négatif	Faux positif
Réel en défaut (1)	Faux négatif	Vrai positif

Equilibre à trouver :

- Optimisation du Recall

$$Recall = \frac{vrais\ positifs}{vrais\ positifs + faux\ négatifs}$$

- Optimisation de la Precision

$$Précision = \frac{vrais\ positifs}{vrais\ positifs + faux\ positifs}$$

Le Recall étant plus important d'un point de vue métier

Fbeta Score

Fbeta Score :
- Compromis entre Recall et Precision

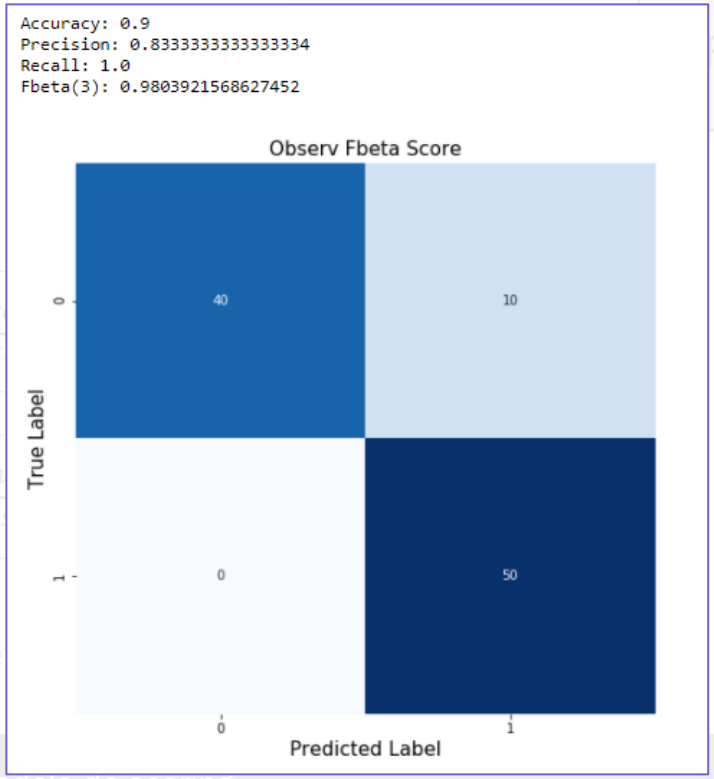
- β correspond à l'importance relative du Recall par rapport à la précision

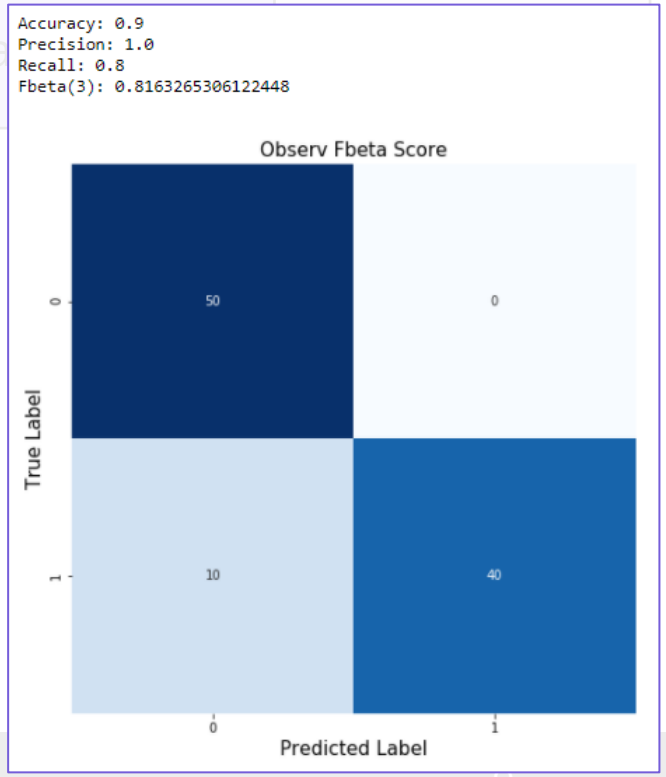
$$F_\beta = (1+\beta^2)\cdot\frac{\text{precision}\cdot\text{recall}}{(\beta^2\cdot\text{precision})+\text{recall}}.$$

In terms of Type I and type II errors this becomes:

$$F_\beta = \frac{(1+\beta^2)\cdot\text{true positive}}{(1+\beta^2)\cdot\text{true positive}+\beta^2\cdot\text{false negative}+\text{false positive}}.$$

- **β = 3** (fixé empiriquement à)
- Fbeta étant compris entre 0 et 1 (1 étant un classifieur parfait)





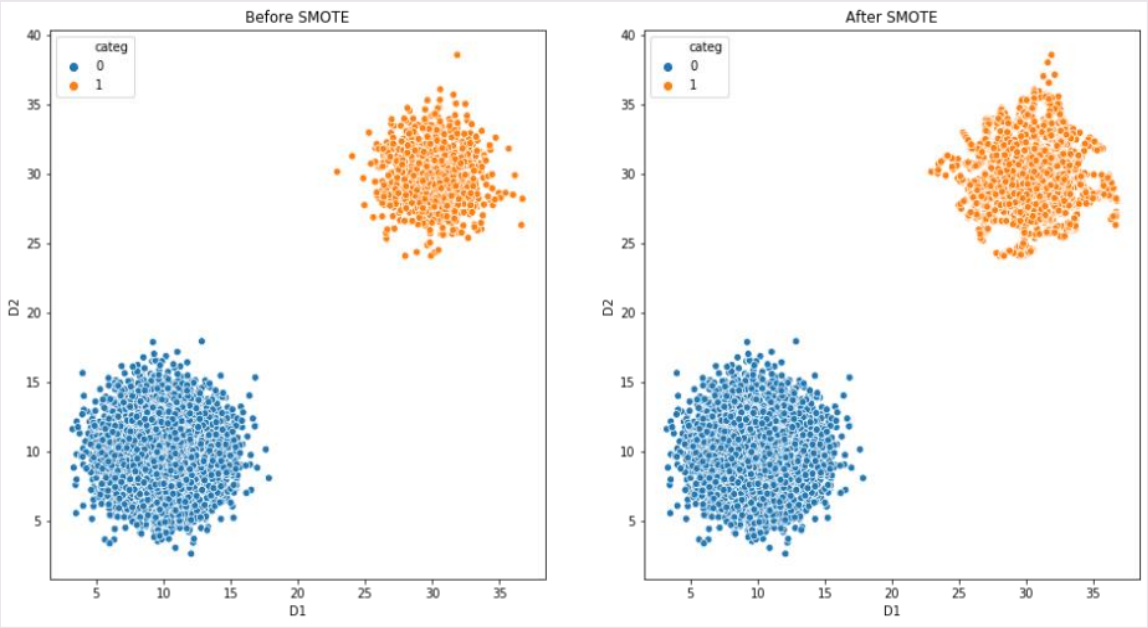
3. Approche de modélisation

Méthodes de gestion du déséquilibre :

- Gestion du déséquilibre en amont :
  - ❑ **Undersampling** : réduit le nombre d'observations de la classe majoritaire au même nombre que la classe minoritaire

  - ❑ **SMOTE** (Oversampling): crée des données synthétiques pour ramener le nombre d'observations de la classe minoritaire au niveau de la classe majoritaire

Perte de données potentiellement intéressantes pour le modèle



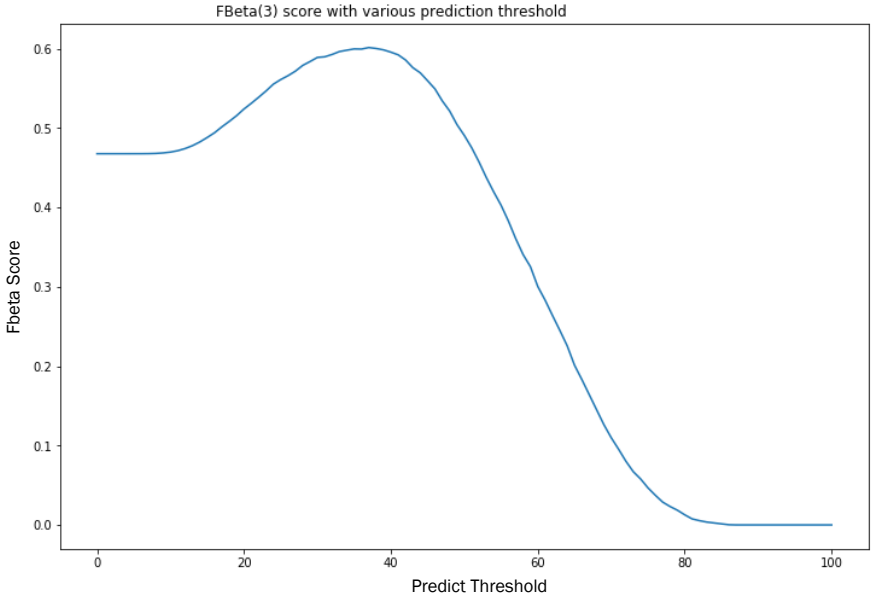
Observation avant et après SMOTE :
- Deux groupes de données avec le même écart-type
- Groupe Bleu 10000 individus
- Groupe Orange initialement 1000 individus (ramené à 10000 via SMOTE)

Conservation d'une distribution locale

3. Approche de modélisation

Méthodes de gestion du déséquilibre :

- Gestion du déséquilibre en amont :
  - ❑ **Undersampling** : réduit le nombre d'observations de la classe majoritaire au même nombre que la classe minoritaire
  - ❑ **SMOTE** (Oversampling): crée des données synthétiques pour ramener le nombre d'observations de la classe minoritaire au niveau de la classe majoritaire
- Gestion du déséquilibre pendant l'entraînement du modèle :
  - ❑ **Class Weight** : appliquer une pénalité plus importante à la fonction de perte lors d'une mauvaise classification de la classe minoritaire
- Gestion du déséquilibre en aval de l'entraînement :
  - ❑ **Gestion du seuil de probabilité** : faire varier le seuil de probabilité à partir duquel on classifie un individu dans la classe 1



3. Approche de modélisation

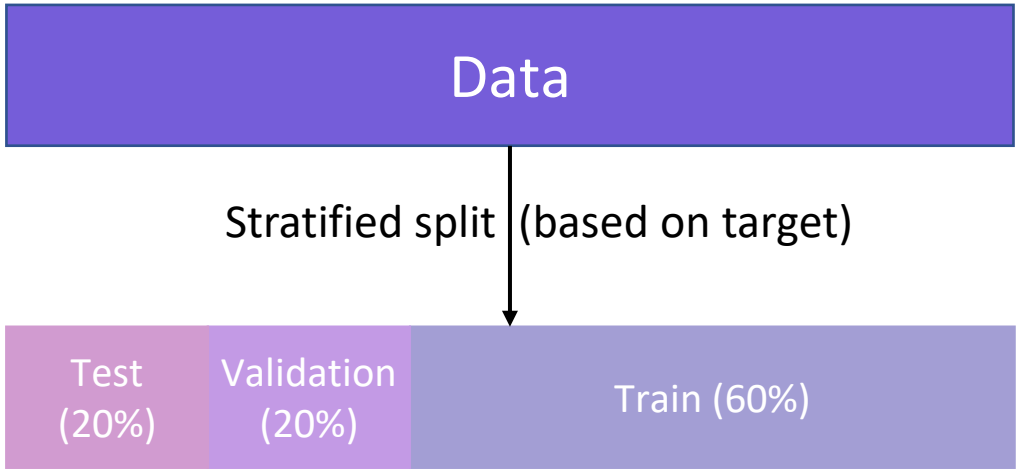
Entraînement et sélection de modèle :

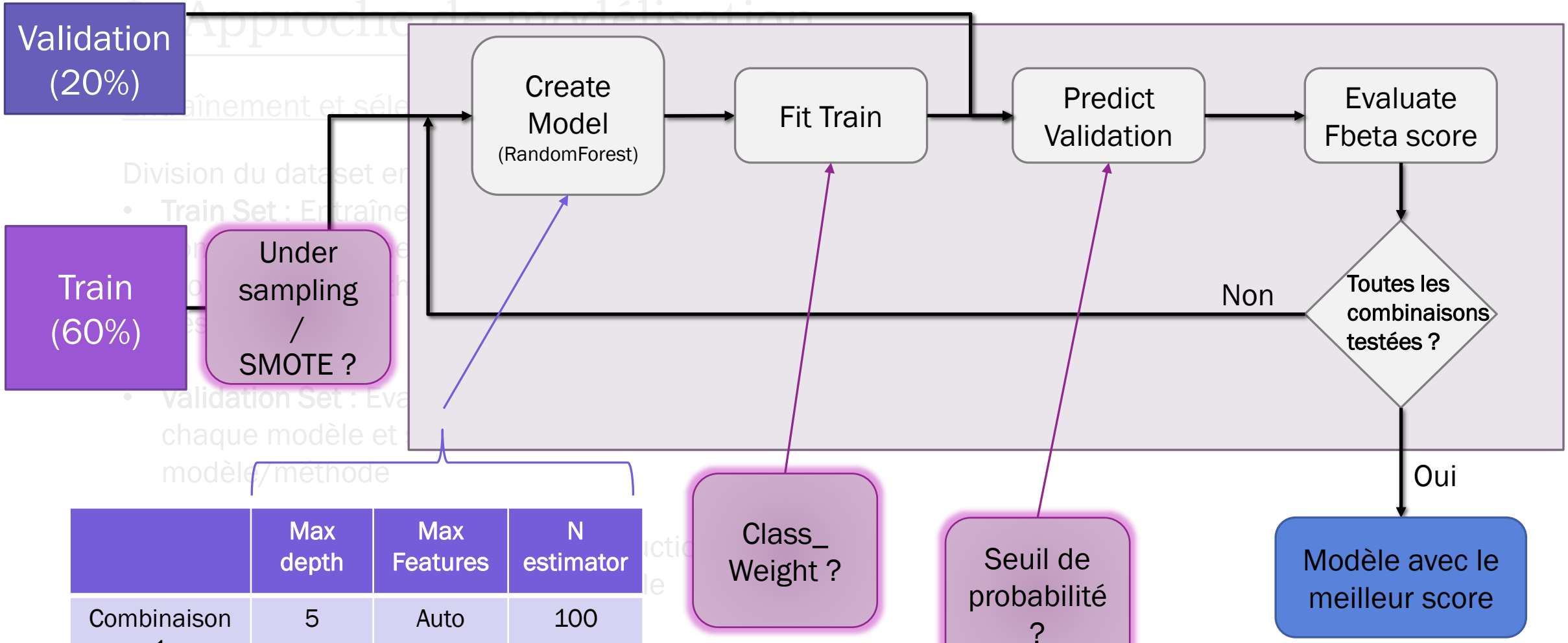
Division du dataset en 3

- **Train Set** : entraîner un modèle pour chaque combinaison d'une grille d'hyperparamètres (pour chaque méthode de gestion de déséquilibre)

- **Validation Set** : évaluation du Fbeta Score de chaque modèle et sélection du meilleur couple modèle/méthode

- **Test Set** : évaluation finale « en production » sur un jeu de données inconnu du modèle

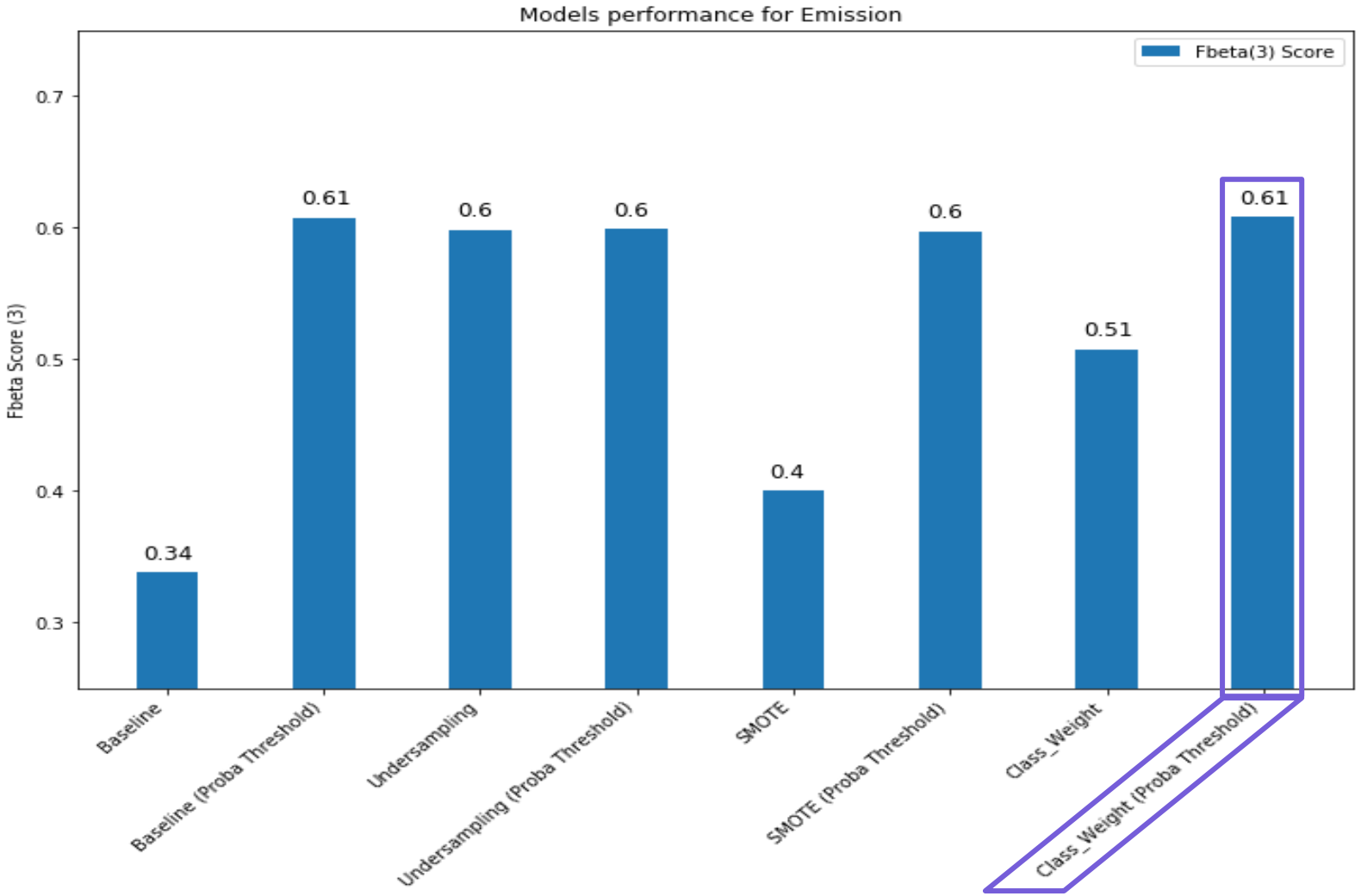


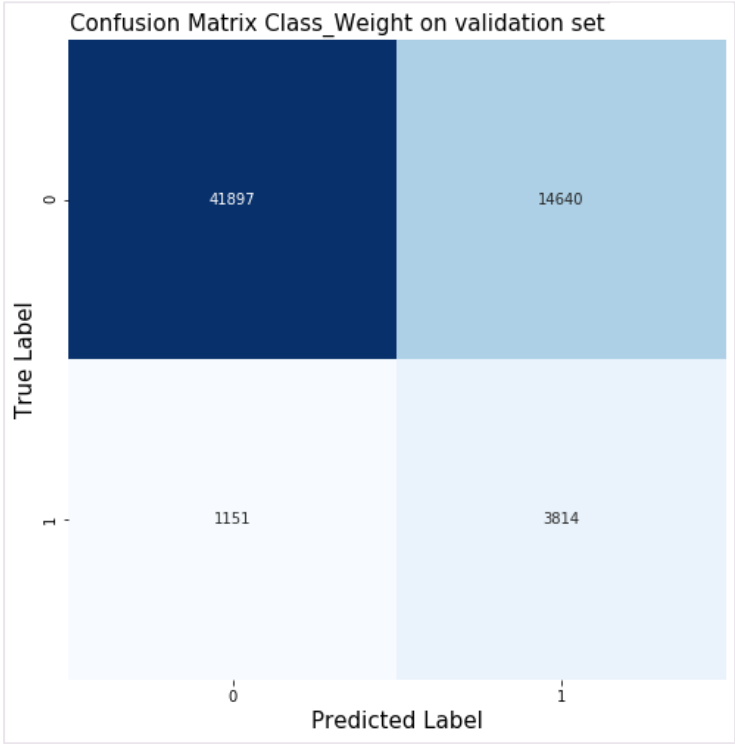


	Max depth	Max Features	N estimator
Combinaison 1	5	Auto	100
Combinaison 2	5	Auto	100
…	…	…	…
Combinaison n	30	Sqrt	200

3. Approche de modélisation

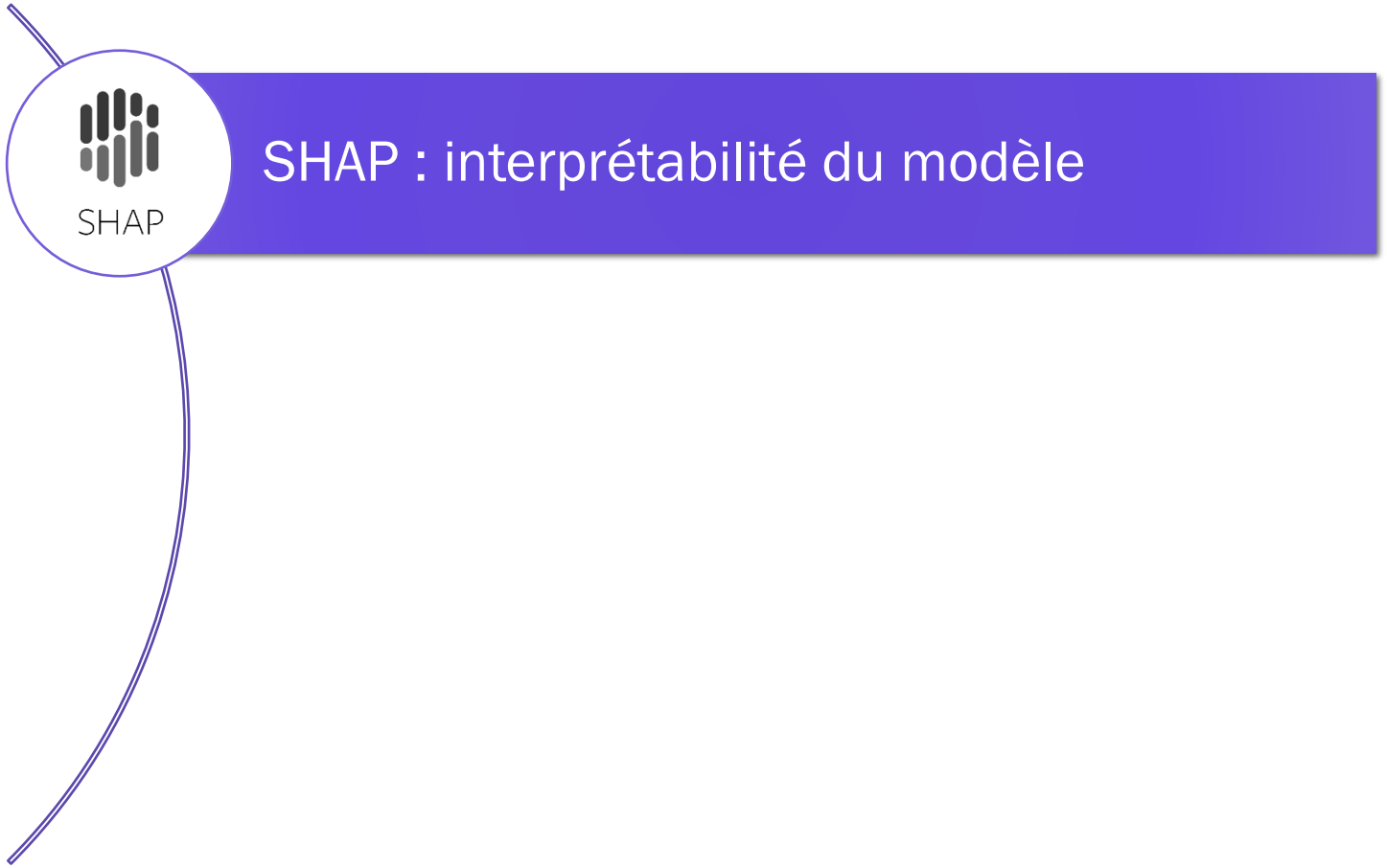
Entraînement et sélection de modèle :





4. Présentation d'un dashboard métier

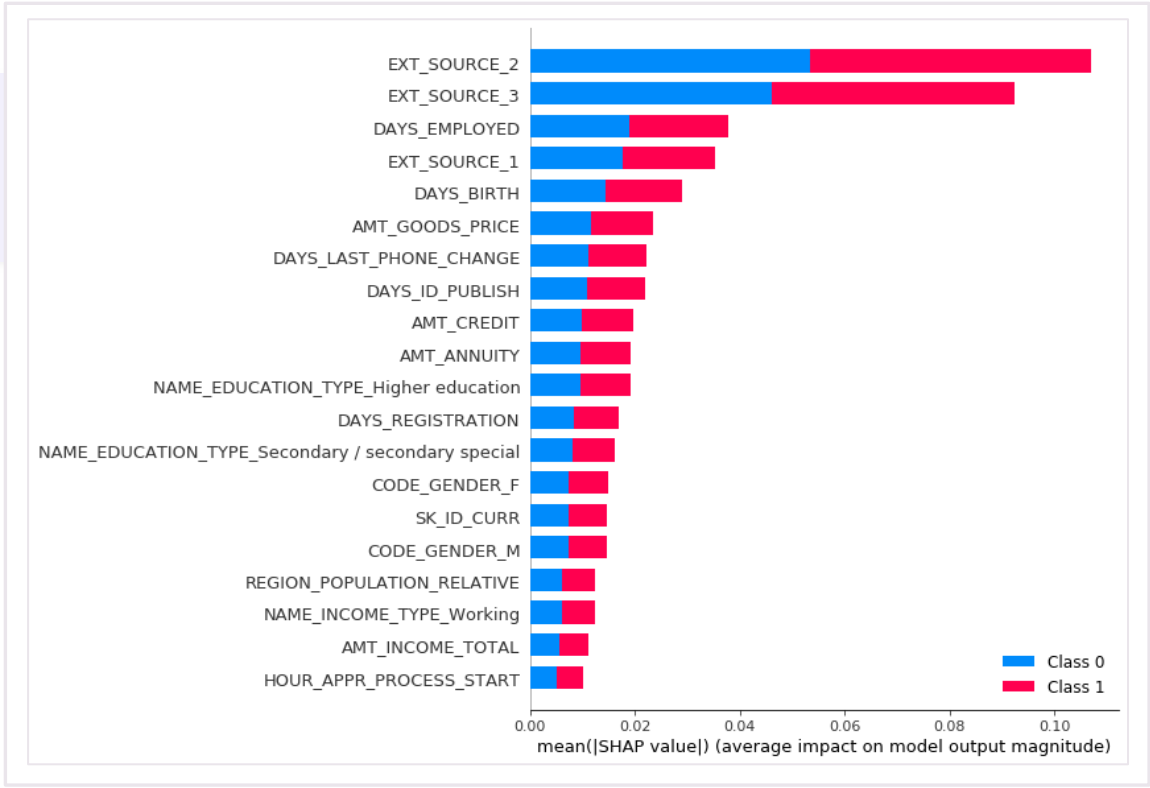
Outils utilisés pour la mise en place du dashboard :



Méthode SHAP

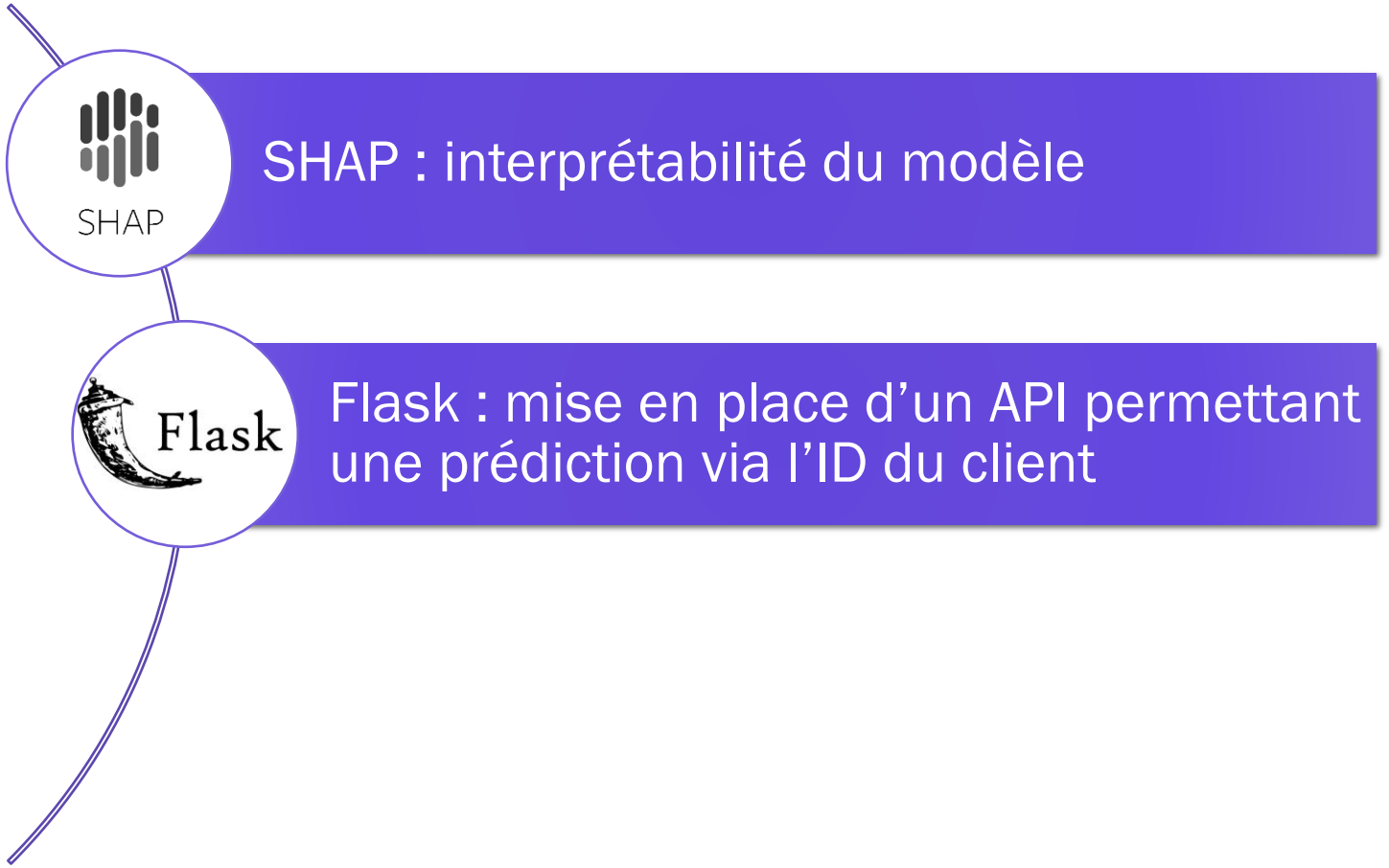
Méthode SHAP :

- Basée sur la théorie des jeux
- Associer à une variable la « moyenne de son impact » pour toutes les combinaisons de variables possibles



4. Présentation d'un dashboard métier

Outils utilisés pour la mise en place du dashboard :



API via Flask

```
(base) C:\Users\quent\Desktop\Formation_OCR\Projets\Projet_7>python app.py
 * Serving Flask app "app" (lazy loading)
 * Environment: production
   WARNING: This is a development server. Do not use it in a production deployment.
   Use a production WSGI server instead.
 * Debug mode: on
 * Restarting with windowsapi reloader
 * Debugger is active!
 * Debugger PIN: 135-139-625
 * Running on http://127.0.0.1:5000/ (Press CTRL+C to quit)
```

Please enter a customer number

(Examples: 217687, 358891, 423598,...)

Customer

ID

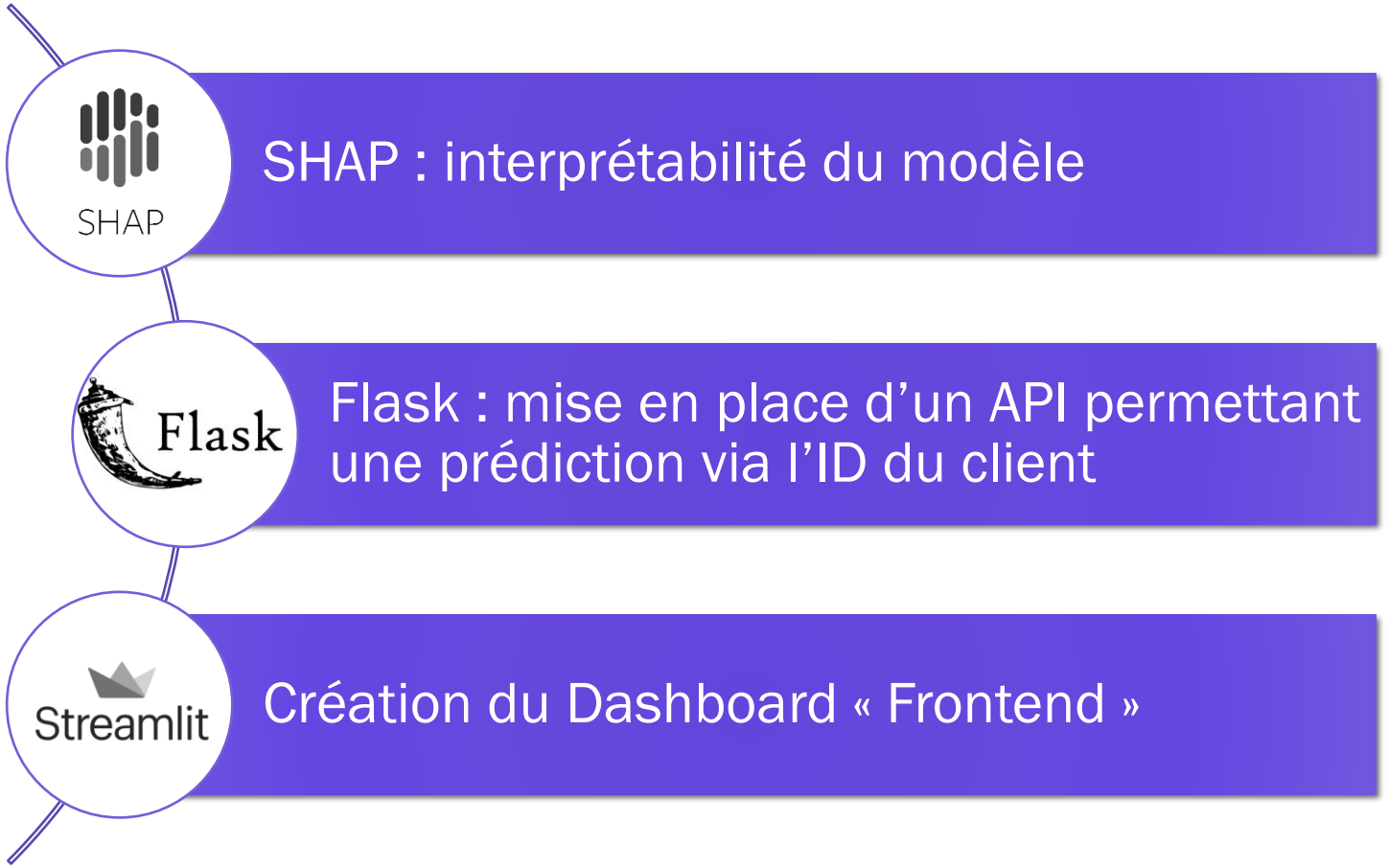
Predict now

127.0.0.1:5000/dashboard/358891


```
{
  "prediction": 0,
  "proba_1": 0.0015384186921438793
}
```

4. Présentation d'un dashboard métier

Outils utilisés pour la mise en place du dashboard :



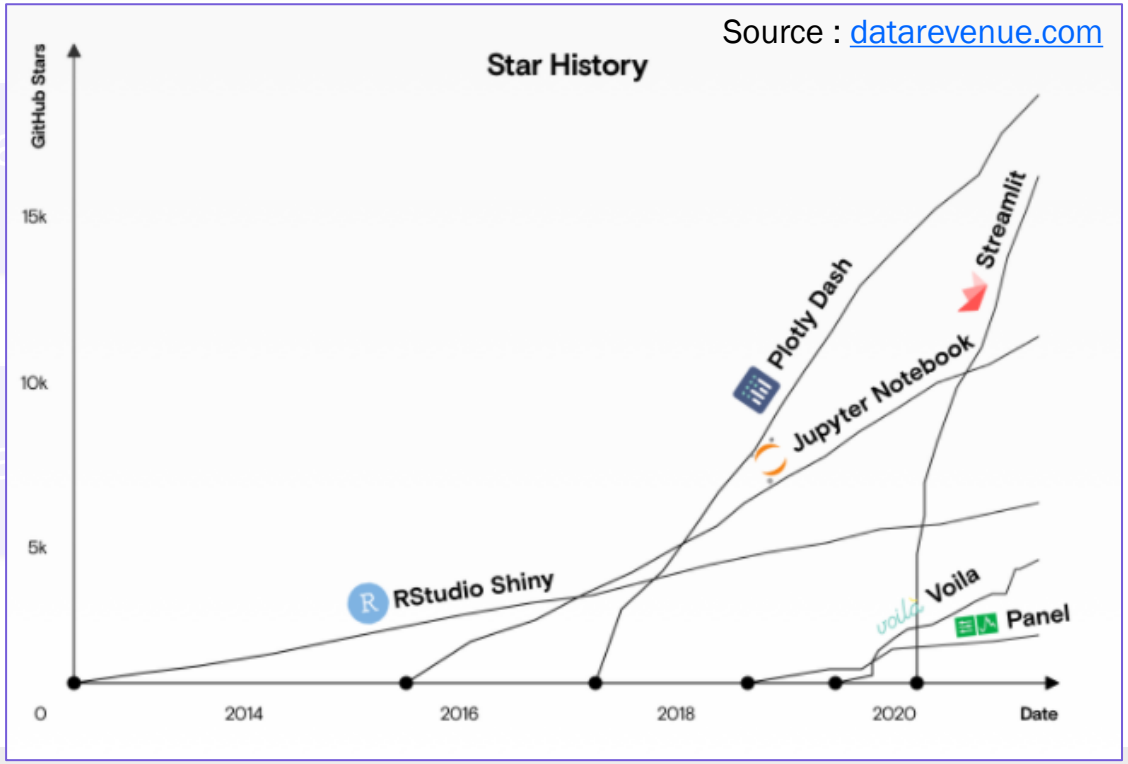
Choix de Streamlit

	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

Streamlit :
- Solution récente est très appréciée
- Simplicité de mise en place
- Solution la plus « plug & play »

Si on souhaite mettre en place un Dashboard un peu plus personnalisé, on pourra se rabattre sur Dash

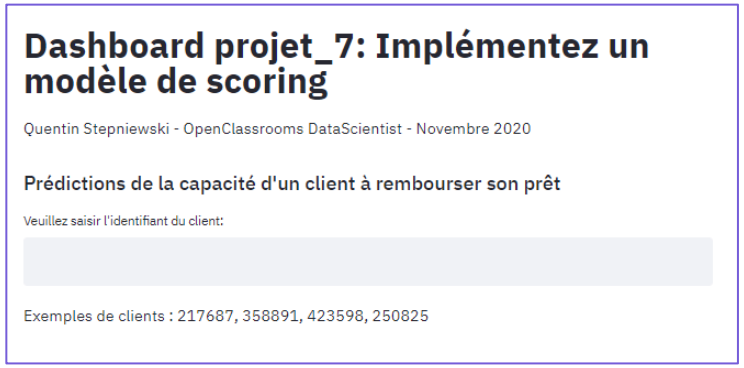
Source : datarevenue.com



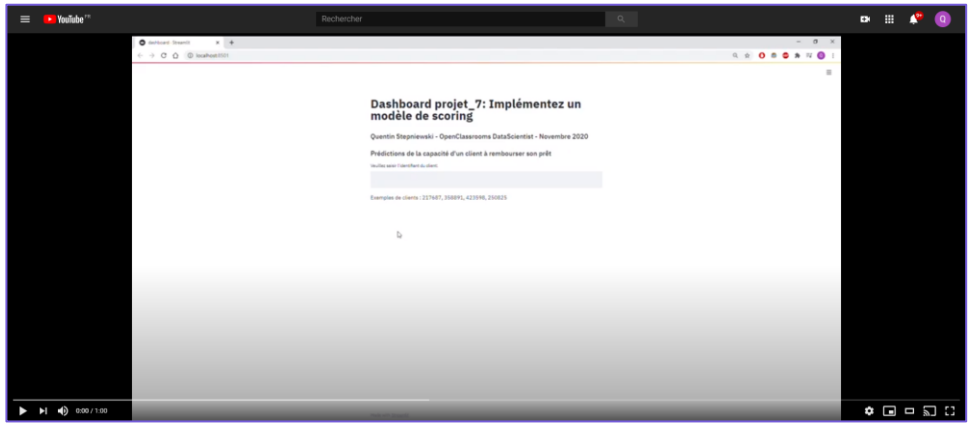
4. Présentation d'un dashboard métier

Présentation du Dashboard

Lien vers le Dashboard



Lien vers la vidéo de présentation



5. Conclusion et mise en perspective

« Proof of concept » établie :

- Modèle de prédiction mis en place
- Dashboard fonctionnel créé

Amélioration possible des performances :

- Affiner la métrique (Fbeta Score) en collaboration avec les équipes métiers (en se basant sur les pertes/coûts d'opportunité réels)
- Augmenter le nombre d'observations pour les individus en défaut
- Mise en place de modèle d'ensemble type Stacking (avec potentiellement une partie dédiée à l'apprentissage des profils avec défaut)
- Amélioration du preprocessing

Amélioration du dashboard :

- Retour des équipes métiers sur l'outil actuel
- Gestion plus spécifique de l'interprétabilité en fonction du besoin

Merci de votre attention

Slides Bonus