



OPENCLASSROOMS

Présentation Projet 7

«Implémentez un modèle
de *scoring*»

Denis Desoubzdanne

9 Décembre 2023

Formation *Data Scientist*

Sommaire

1. Présentation du sujet
2. Jeu de données et *pre-processing*
3. Explication de l'approche de modélisation
4. Déploiement du modèle *via* une API sur le web
5. Démonstration du *dashboard*
6. Conclusion et bilan

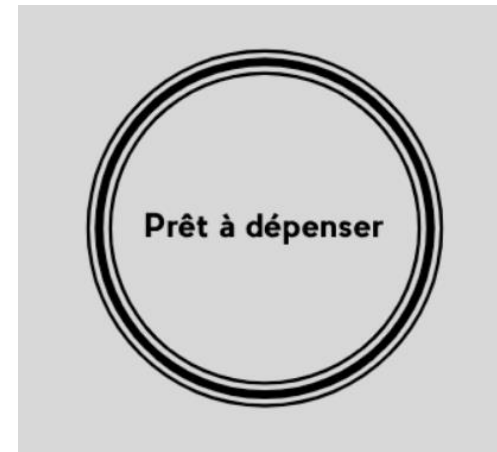
01

Présentation du sujet

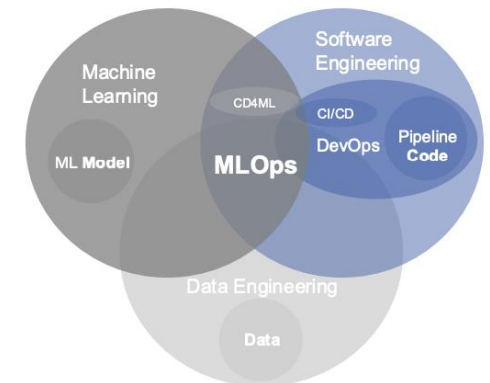
Objectif, missions

Data Scientist/MLOps

Société financière “Prêt à dépenser” souhaitant mettre en oeuvre un outil de “scoring credit” pour calculer la probabilité qu’un client rembourse son crédit



- Mon objectif : **Développer un modèle** de classification et le déployer sur une **plateforme web**
- Mes missions :
 - Construire un **modèle de scoring** (prédiction sur la probabilité de faillite)
 - Construire un **dashboard interactif** (interprétation des prédictions)
 - Mettre **en production** : le modèle (à l’aide d’une **API**) et le **dashboard**



02

Jeux de données

Overview, pre-processing & déséquilibre

Overview

- 1 dossier **‘.zip’** à télécharger

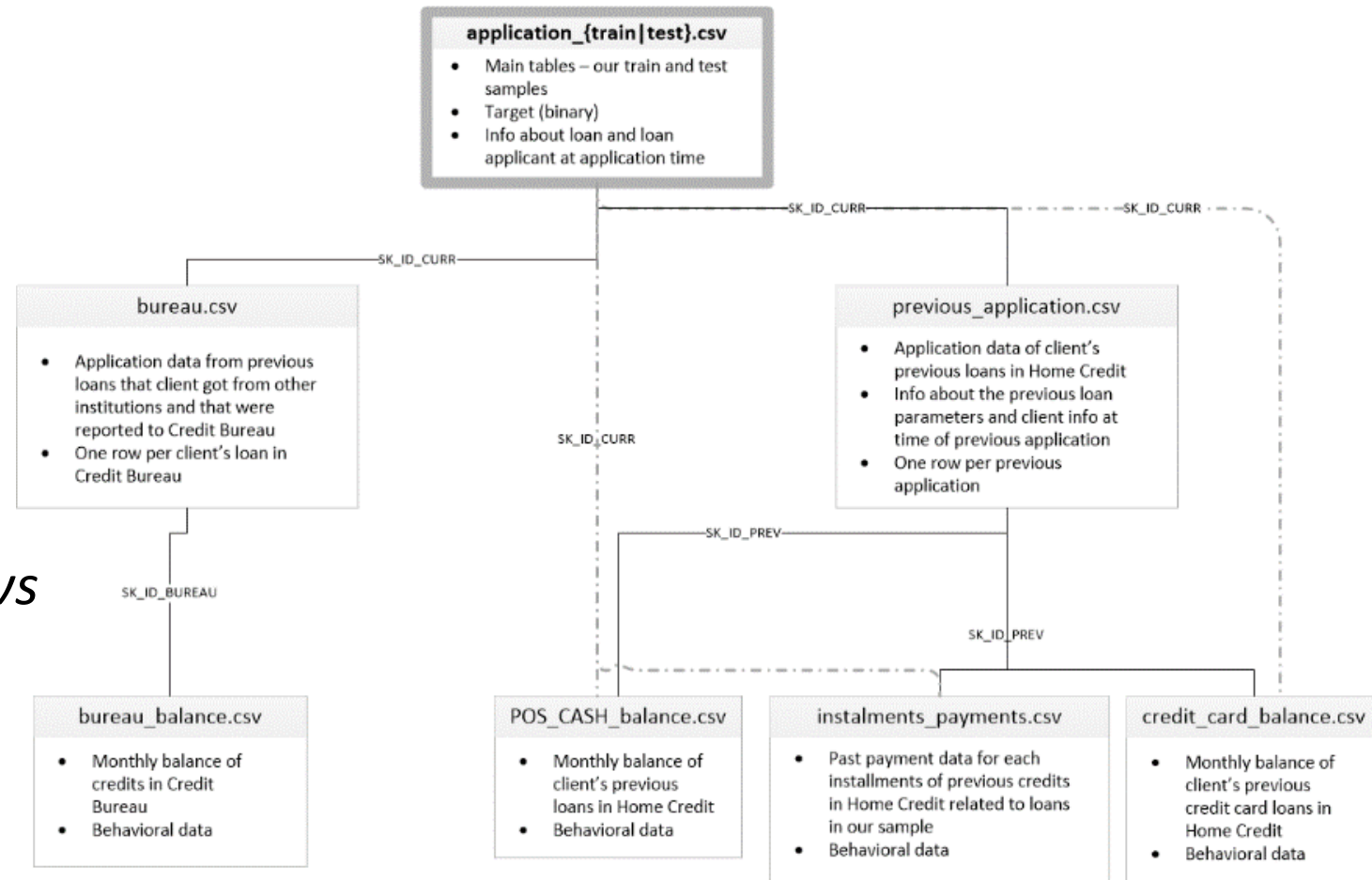
➤ 10 fichiers **‘.csv’**

➤ Jeu de données **Kaggle : « Home Credit Default Risk »**

- Données clients : 307511 (*training*) vs 48744 (*test*)

- **122 features** (au départ) : âge, sexe, emploi, logement, revenus...

➤ **feature cible (« TARGET »)** : « 0 » = solvable et « 1 » : non solvable



Pre-processing

- Récupération d'un kernel sous **Kaggle** :

=> *Notebooks* de Will Koehrsen (<https://www.kaggle.com/willkoehrsen>)



=> EDA, préparation des données et *features engineering*

- **Adapation** des scripts à notre problématique :

=> *One-hot encoding*

=> Détection des *outliers*/anomalies

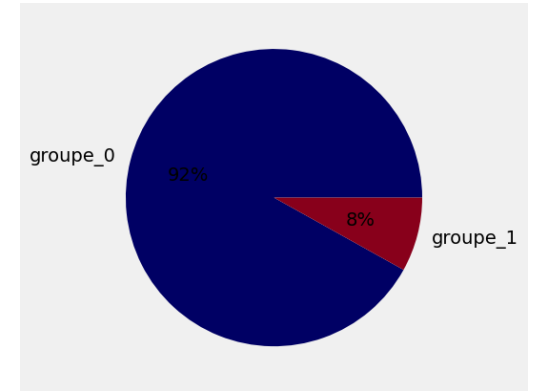
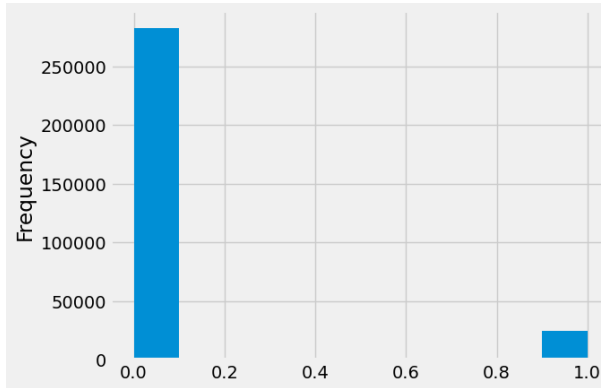
=> Création de *features* métiers : durée de crédit, ratio montant du crédit sur revenu, ratio annuités sur revenu...

=> Imputation des valeurs manquantes

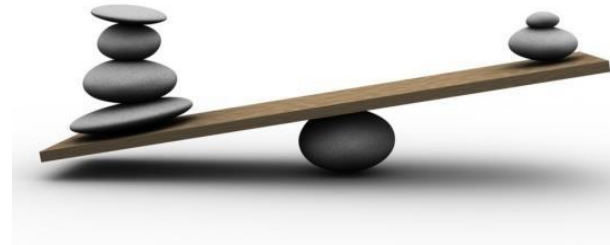
- Nouveaux jeu de données de **1244 features**

Jeu de données déséquilibré

- **92 % des clients solvables** (classés « 0 ») vs
- **8 % des clients non solvables** (classés « 1 »)



- Fort déséquilibre du jeu de données!



- **Modèle naïf** (tous les individus sont solvables) : 92% d'exactitude (« *accuracy* »)

03

Explication de l'approche de modélisation

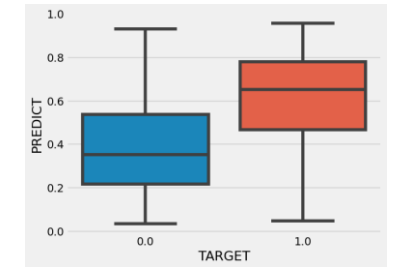
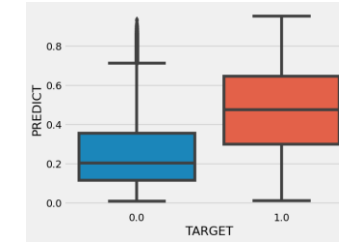
*Gestion du déséquilibre,
Entraînement du modèle,
Score métier*

Comment réduire les conséquences du déséquilibre?

- Collecter plus de données clients non solvables (classe 1)
- Création d'individus « artificiels » (SMOTE)
- ★ • ***Under-sampling*** (réduire le nombre d'individus sur-représentés) = Méthode 2
- ★ • **Pondération** des observations dans le *training* = Méthode 1
- ★ • Choix d'une **métrique de performance « customisée »**

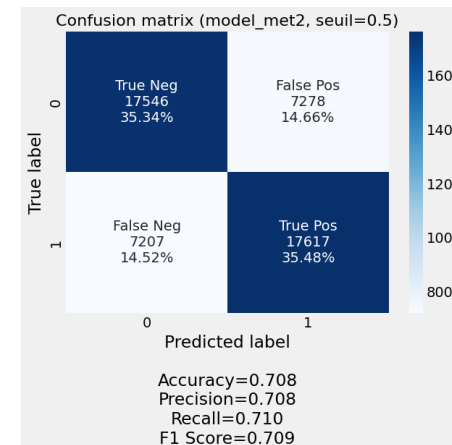
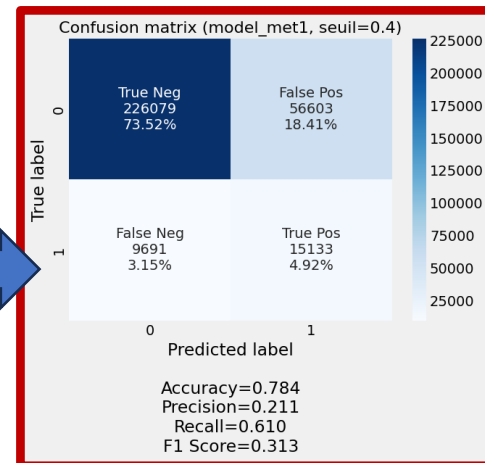
Méthode choisie

- Algorithme choisi : LGBMClassifier
- **Méthode 1** : $class_weight = \{0: 0.15, 1: 0.85\}$
- Méthode 2 : *under-sampling*



=> *comparaison : selon l'exactitude (TP+TP/all)*

FN = 3,15%



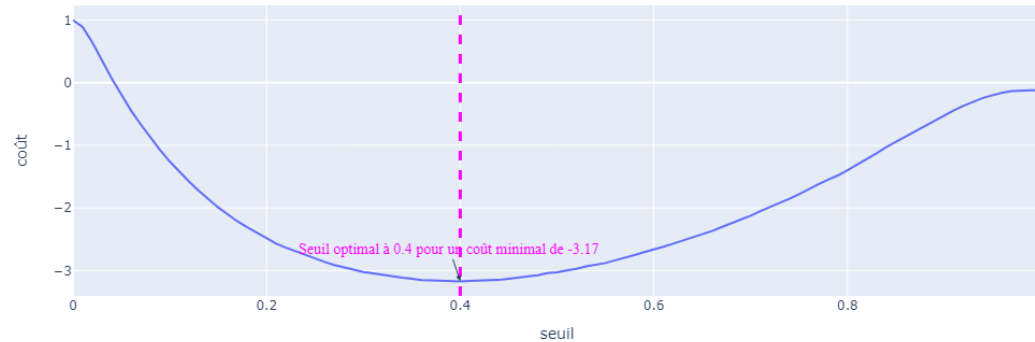
Optimiser le seuil de prédiction

- Création d'une fonction « coût »

$$Coût = \frac{100*FN - 10*TN + 1*(TP+TN+FP+FN)}{TP+TN+FP+FN}$$

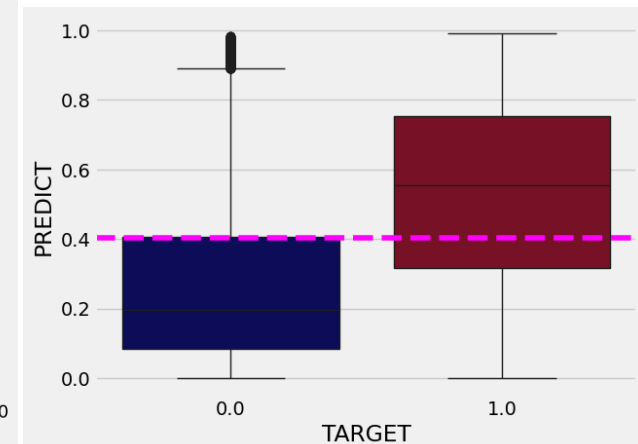
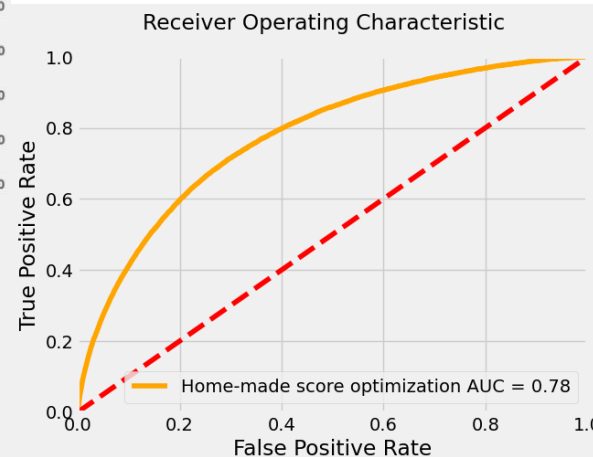
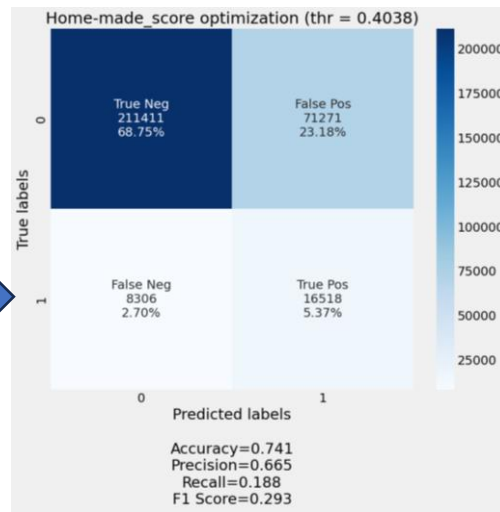
⇒ **minimiser les FN**

Coût normalisé en fonction du seuil



Objet	Coût par client (unité arbitraire)	Classe
Octroi de crédit à un client qui fait défaut	100	FN (False Negative)
Octroi de crédit à un client qui ne fait pas défaut	-10	TN (True Negative)
Refus de crédit à un client qui aurait fait défaut	0	TP (True Positive)
Refus de crédit à un client qui n'aurait pas fait défaut	0	FP (False Positive)
Frais généraux pour chaque client	1	-

⇒ résultats



Seuil_opt = 0,4

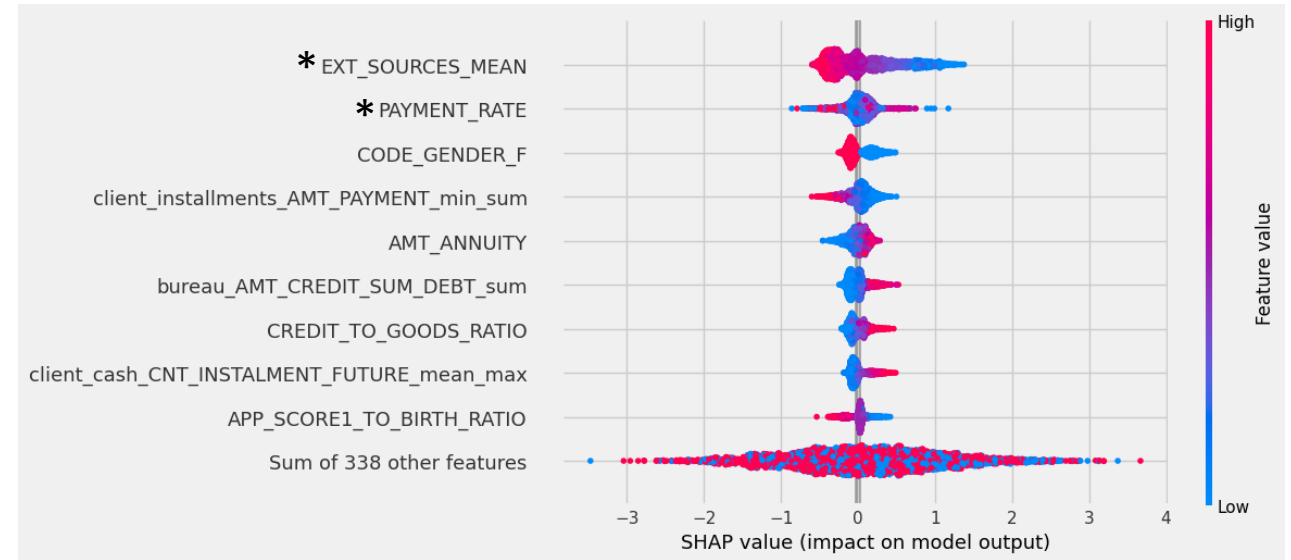
Interprétabilité du modèle

- Globale : *SHAP values*

**Ext_Sources_Mean* = Moy des 3 sources de revenus

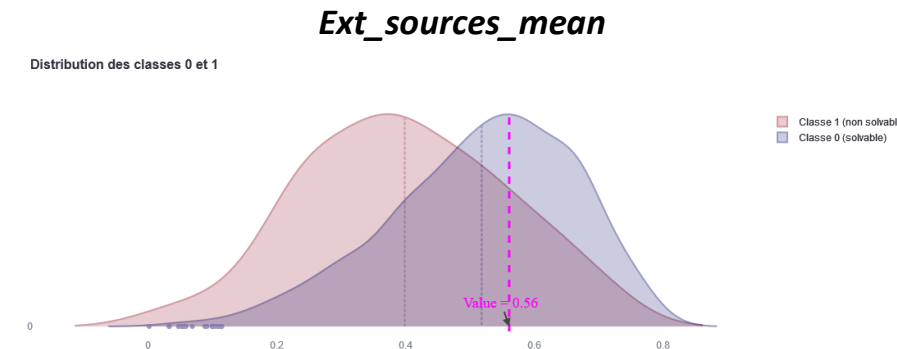
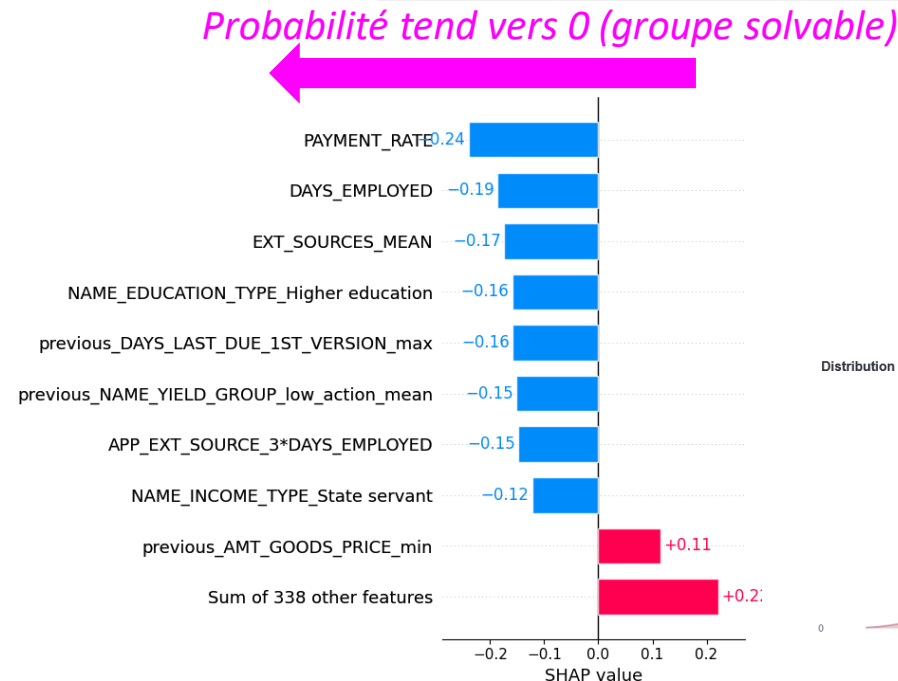
**Payment_Rate* = $AMT_Annuity / AMT_Credit$

AMT_CREDIT	Credit amount of the loan
AMT_ANNUITY	Loan annuity























- Locale : *SHAP values*

**Ex : individu n°204226
(classé 0)**



Analyse du *Data Drift* : 35% des feat.

- Librairie : *evidently*

Data Drift Summary						
Drift is detected for 35.159% of columns (122 out of 347).						
Column	Type	Reference Distribution	Current Distribution	Data Drift	Stat Test	Drift Score
> DAYS_BIRTH	num			Detected	Wasserstein distance (normed)	7.372493
> APP_DAYS_EMPLOYED_DAYS_BIRTH_diff	num			Detected	Wasserstein distance (normed)	5.901742
> APP_SCORE1_TO_BIRTH_RATIO	num			Detected	Wasserstein distance (normed)	5.449058
> APP_SCORE2_TO_BIRTH_RATIO	num			Detected	Wasserstein distance (normed)	4.379812
> APP_SCORE3_TO_BIRTH_RATIO	num			Detected	Wasserstein distance (normed)	4.352126
> ID_TO_BIRTH_RATIO	num			Detected	Wasserstein distance (normed)	3.926454
> INCOME_TO_BIRTH_RATIO	num			Detected	Wasserstein distance (normed)	3.093723
> APP_EXT_SOURCE_2*EXT_SOURCE_3*DAYS_BIRTH	num			Detected	Wasserstein distance (normed)	2.889899
> PHONE_TO_BIRTH_RATIO	num			Detected	Wasserstein distance (normed)	2.728779
> DAYS_EMPLOYED_PERC	num			Detected	Wasserstein distance (normed)	2.306072
10 rows 1-10 of 347						



Data drift observé sur des features dépendantes de l'âge du client

04







Déploiement du modèle *via* une API sur le Web

Entrainement d'un modèle et suivi de performance (mlflow)

Création et déploiement d'une API sur le web (+tests unitaires)

Création et déploiement d'un dashboard sur le web

Outils utilisés (MLOps)



Solution	Description
 MLFlow	Plateforme pour faire du déploiement continu de modèles de ML (<i>tracking</i>)
 SHAP	Librairie utiliser pour une meilleure interprétabilité (locale ou globale) des <i>features</i> d'un modèle
 FastAPI	API permettant d'appeler la prédiction à partir de l'ID d'un client (en locale ou hébergé) : <i>back-end</i>
 Streamlit	Outil gratuit pour réaliser des tableaux de bord : <i>front-end</i>
 GitHub	Plateforme de <i>versioning</i> et de déploiement continu (approche CI/CD) de scripts
 Microsoft Azure	Plateforme informatique <i>Cloud</i> pour le déploiement d'applications notamment



Ré-entraînement du modèle et *tracking* (mlflow)



mlflow 2.5.0 Experiments Models GitHub Docs

Experiments

Search Experiments

☐ Default  

☒ Best_LGMClassifier_final  

☐ Best_LGMClassifier  

Best_LGMClassifier_final

Experiment ID: 345067922943021846 Artifact Location: file:///c:/Users/denis/OneDrive/Documents/Denis/Formation_OpenClassRooms/Projet7_DDDesoubzdanne/Desoubzdanne_Denis_2_dossier_code_112023/mlruns/345067922943021846












Provide Feedback

Share

Description Edit

Table view Chart view Artifact view Time created State Active Refresh

Sort: Created Columns Expand rows

<input type="checkbox"/>		Run Name	Created	Dataset	Duration	Source	Models
<input type="checkbox"/>		 Best_LGMClassifier_final	 22 days ago	-	11.0s	C:\Users\...	 sklearn
<input type="checkbox"/>		 Best_LGMClassifier_final	 22 days ago	-	283ms	C:\Users\...	-
<input type="checkbox"/>		 Best_LGMClassifier_final	 22 days ago	-	313ms	C:\Users\...	-

3 matching runs

> Datasets

> Parameters (8)

> Metrics (8)

Name	Value
Accuracy	0.741
Auc_score	0.776
F1_score	0.293
Fit_time	74.25
Home-made_score	-3.174
Precision	0.665
Recall	0.188
Threshold	0.404

API, tests unitaires et *dashboard* en local

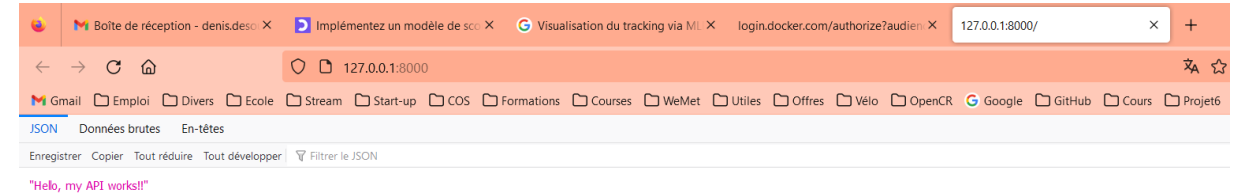
- API running (local host)



- Tests unitaires (local)



- Dashboard running (local)



```
raise_custom_warning()

# Replace with actual data for prediction
base_url = "http://127.0.0.1:8000"
endpoint = "/prediction/"
url = f"{base_url}{endpoint}"

try:
    # Send request for usecase1
    response = requests.get(url, data=data1, timeout=80)
    response.raise_for_status() # Raise an HTTPError for bad responses
    proba_default = eval(response.content)["probability"]
    result = round(proba_default * 100, 1)

    # Send request for usecase0
    response0 = requests.get(url, data=data0, timeout=80)
    response0.raise_for_status()
    proba_default0 = eval(response0.content)["probability"]
    result0 = round(proba_default0 * 100, 1)

    # Assertions
    assert response.status_code == 200
    assert "probability" in response.json()
    assert result >= 40 # threshold 40% for usecase1
    assert result0 < 40 # threshold 40% for usecase0

except requests.exceptions.RequestException as e:
    # Handle exceptions, print an error message, or fail the test
    print(f"Request failed: {e}")
```

/!\ tester que l'API renvoie bien une probabilité à partir du modèle (tests sur 2 json files : client 0 (au-dessus du seuil) et client 1 (en-dessous du seuil))

```
> resources
- dashboard.py 9+
- README.md U
- requirements.txt
- Desoubzdanne_Denis_2_dossier_code_122023
- jupyter_checkpoints
- mirrors
- col_transfo.joblib
- ddrift_report.html
- Desoubzdanne_Denis_2_notebook_112023_1ere...
- Desoubzdanne_Denis_2_notebook_122023.ipynb
- input0.json
- input1.json
- test_samp.csv
- test2.csv
- train_samp.csv
- train2.csv
- Desoubzdanne_Denis_3_note_methodologique_12...
- Desoubzdanne_Denis_4_presentation_122023.pptx
- README.md

41 ## DECLARING ALL THE FUNCTIONS NEEDED ##
42 #HOSTING
43 def host(local:bool):
44     if local is True:
45         HOST = 'http://127.0.0.1:8000'
46     else:
47         HOST = 'https://basicwebappv1.azurewebsites.net/' #Azure
48     return HOST
49
50 HOST = host(local=False)
51
52 #GET LIST OF IDS
53 def get_ids():
54     try:
55         response = requests.get(HOST+"/get_ids/")
56         response.raise_for_status() # check for HTTP errors
57         ids = eval(response.content)["data"]
58         return ids
59     except requests.RequestException as e:
60         print(f"Error fetching data from API: {e}")
61
62 list_ids = get_ids()
63
64 #GET LIST OF FEATURES
65 def get_feat():
```

Projet 7 : 'Implémentez un modèle de scoring'

Sélection du client par son numéro 0



Versioning des scripts sous GitHub

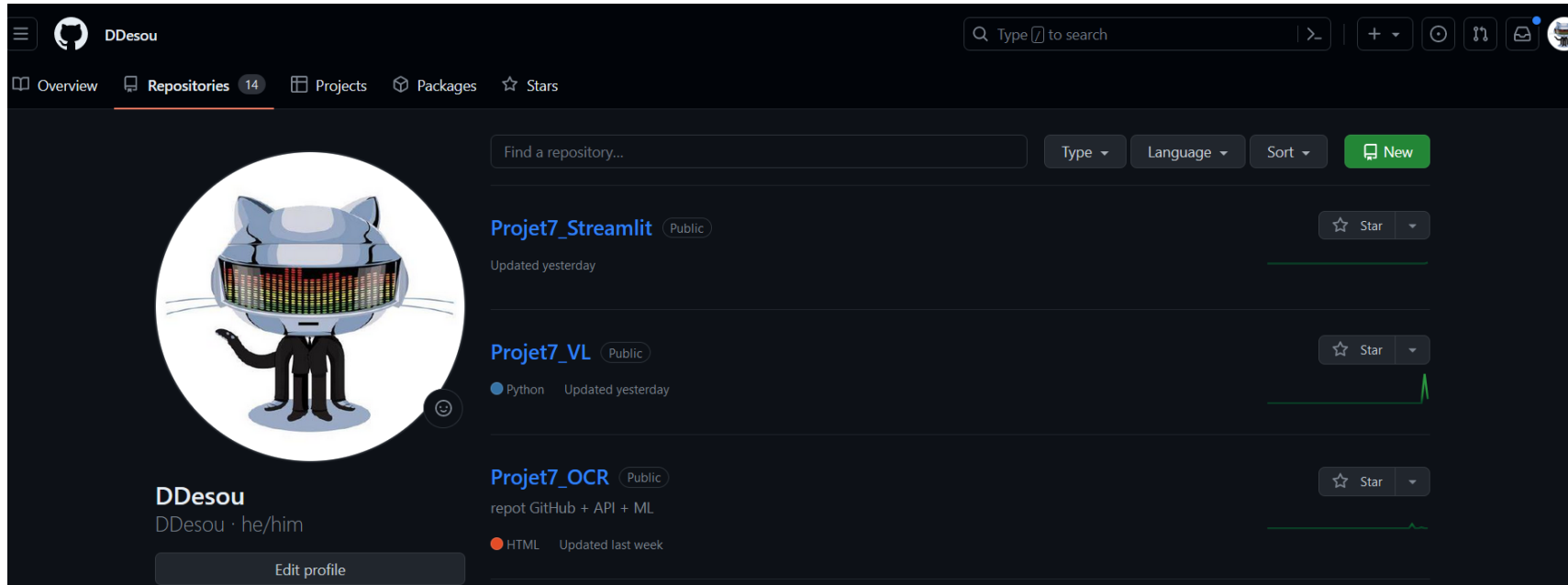
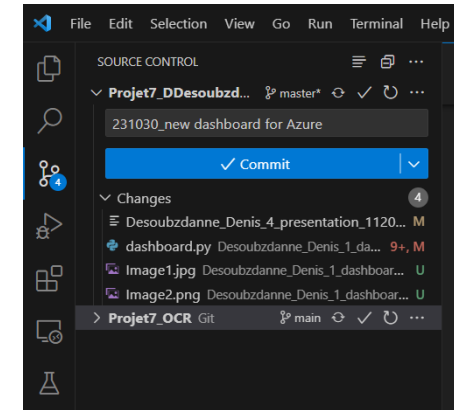


Tableau de bord sous VSCode

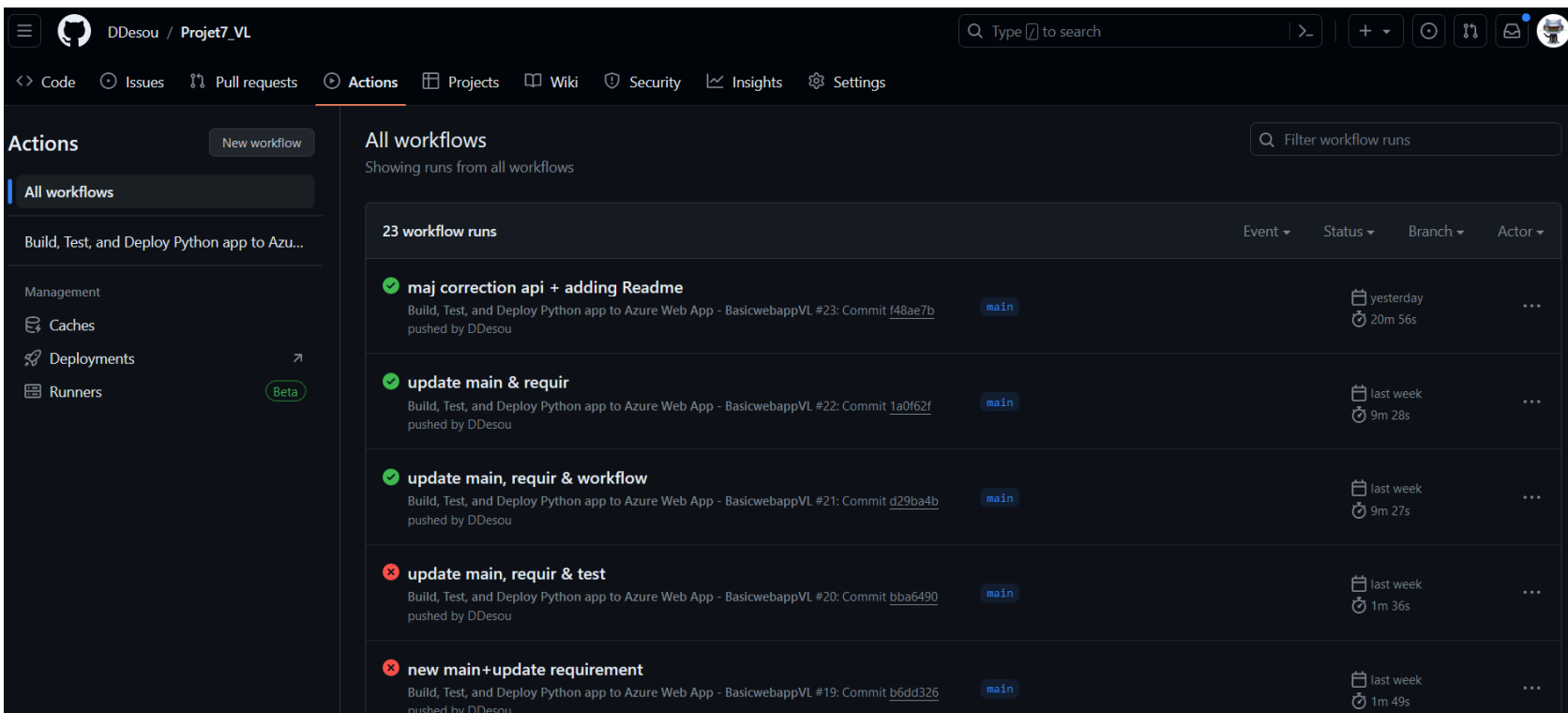


- 3 repositories :

- *Projet7_VL => repos API (https://github.com/DDesou/Projet7_VL.git)*
- *Projet7_Streamlit => repos dashboard (https://github.com/DDesou/Projet7_Streamlit.git)*
- *Projet7_OCR => repos codes et livrables (https://github.com/DDesou/Projet7_OCR.git)*

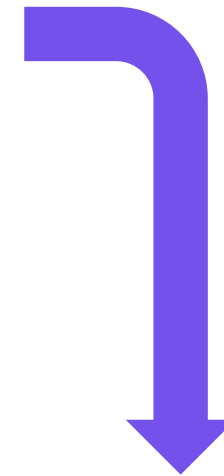
- *Git add .*
- *Git commit -m 'my_message'*
- *Git push*

Déploiement continu (CI/CD) : GitHub Actions

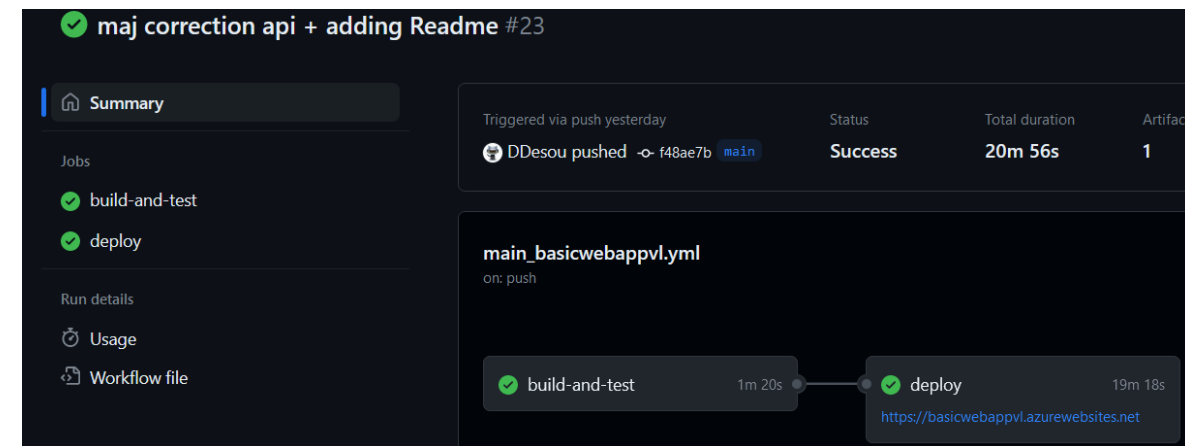


The screenshot shows the GitHub Actions interface for the repository 'DDesou / Projet7_VL'. The 'Actions' tab is selected, displaying a list of workflow runs under the heading 'All workflows'. The list shows 23 workflow runs. The first five runs are visible, with their status, event, branch, and actor. The first run, 'maj correction api + adding Readme', is successful and was triggered by a push to the 'main' branch by user 'DDesou' yesterday, taking 20m 56s. The second run, 'update main & requir', is also successful and was triggered by a push to the 'main' branch by 'DDesou' last week, taking 9m 28s. The third run, 'update main, requir & workflow', is successful and was triggered by a push to the 'main' branch by 'DDesou' last week, taking 9m 27s. The fourth run, 'update main, requir & test', failed and was triggered by a push to the 'main' branch by 'DDesou' last week, taking 1m 36s. The fifth run, 'new main+update requirement', failed and was triggered by a push to the 'main' branch by 'DDesou' last week, taking 1m 49s.

Workflow Run	Status	Event	Branch	Actor
maj correction api + adding Readme	Success	push	main	DDesou
update main & requir	Success	push	main	DDesou
update main, requir & workflow	Success	push	main	DDesou
update main, requir & test	Failure	push	main	DDesou
new main+update requirement	Failure	push	main	DDesou



/!\ Tests unitaires (Pytest) directement intégrés dans le déploiement continu



The screenshot shows the detailed view of the workflow run 'maj correction api + adding Readme #23'. The 'Summary' tab is selected, showing the run was triggered via push yesterday by 'DDesou' to the 'main' branch, with a status of 'Success', a total duration of 20m 56s, and 1 artifact. The workflow file is 'main_basicwebappvl.yml' and it runs on 'push'. The run details show two jobs: 'build-and-test' (1m 20s) and 'deploy' (19m 18s), both of which are successful. The 'deploy' job includes a link to the deployment URL: <https://basicwebappvl.azurewebsites.net>.

Job	Status	Duration
build-and-test	Success	1m 20s
deploy	Success	19m 18s

Création de *web apps* sous Azure

Azure services



Create a resource



Cost Management ...



App Services



Resource groups



App registrations



Subscriptions



Static Web Apps



Monitor



Container instances



More services

Resources

Recent

Favorite

Name

Type

Last Viewed



BasicwebappVL

App Service

26 minutes ago



MyStreamlit

App Service

27 minutes ago



rg-projet7

Resource group

5 days ago



Azure subscription 1

Subscription

2 weeks ago

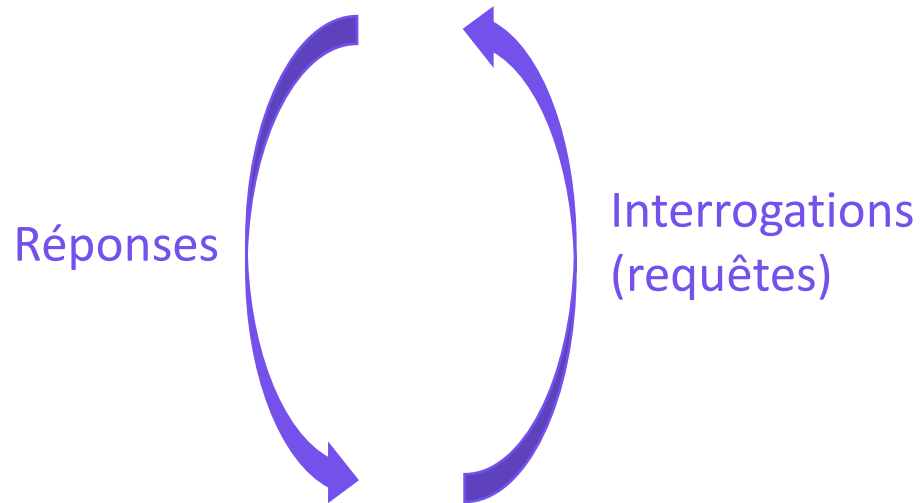
[See all](#)

Back-end (API)
Front-end (dashboard)

/!\ Connectés aux différents repos

Deploiement sous Azure

- API
(<https://basicwebappvl.azurewebsites.net/>)



- Dashboard
(<https://mystreamlit.azurewebsites.net/>)

Microsoft Azure

BasicwebappVL

Overview

Essentials

Resource group (move) : rg-projet7

Status : Running

Location (move) : France Central

Subscription (move) : Azure subscription_1

Subscription ID : 06b74d93-ffa5-4415-9d97-51b6f6702be1

Tags (edit) : myapp:ok

Properties

Web app

Name : BasicwebappVL

Publishing model : Code

Runtime Stack : Python - 3.11

Domains

Default domain : basicwebappvl.azurewebsites.net

Custom domain : Add custom domain

Deployment Center

Deployment logs

Last deployment : Successful on mardi 5 décembre, 03:18:47 AM Refresh

Deployment provider : GitHubAction

Microsoft Azure

MyStreamlit

Overview

Essentials

Resource group (move) : rg-projet7

Status : Running

Location (move) : France Central

Subscription (move) : Azure subscription_1

Subscription ID : 06b74d93-ffa5-4415-9d97-51b6f6702be1

Tags (edit) : mydash:working

Properties

Web app

Name : MyStreamlit

Publishing model : Code

Runtime Stack : Python - 3.11

Domains

Default domain : mystreamlit.azurewebsites.net

Custom domain : Add custom domain

Deployment Center

Deployment logs

Last deployment : Loading deployments... Refresh

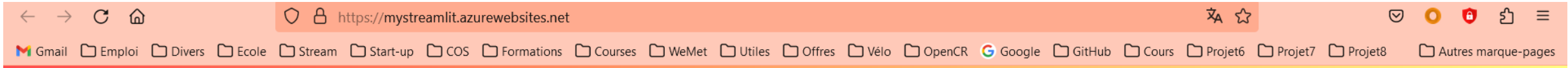
Deployment provider : GitHubAction

05

Démonstration du *dashboard*

Streamlit déployé sous Azure

Dashboard interactif (en ligne)



Projet 7 : 'Implémentez un modèle de scoring'

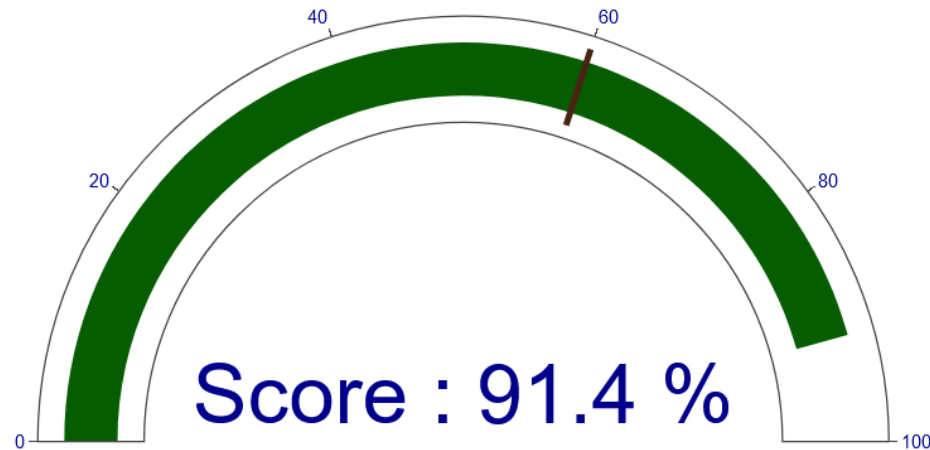


Sélection du client par son numéro ID

351831



Probabilité du client 351831 d'être solvable (classe 0))



06

Conclusion & bilan

Conclusion & bilan

- Mise au point d'un modèle de classification

➤ *LightGBM Classifier*  LightGBM

➤ *Gestion du déséquilibre des groupes*



- Création d'un *scoring* basé sur une fonction de 'coût métier'



- Interprétabilité globale et locale par les **SHAP values** et évaluation du **data drift**

➤ *Modèle (jeu de données) imparfait(s)*



- Utilisation de *mlflow* pour le *tracking* du modèle



Streamlit

- Création d'une API et d'un dashboard sur le web



- *Versioning* des codes + déploiement continu (approche CI/CD)



DATA SCIENCE





OPENCLASSROOMS

Denis Desoubzdanne

9 Décembre 2023

Formation *Data Scientist*

